Technical Information

ASTMAC エンジニアリングガイド

TI 34P02K02-01

はじめに

■ 本書について

本書は，ASTMAC を使用したシステム構築を行う際のエンジニアリングガイドです。ASTMAC を知らない方でも，本書を一読して頂くことによりエンジニアリングすることが可能になります。一貫して生産管理システムを例として記述しておりますので，具体的にイメージを掴むことができます。
また，本書の例に習って内容を埋め込むことにより，機能設計書，基本設計書等の書類が作成できます。
では，ASTMAC でシステムを作成する場合の全体の流れと作成できる書類を示します。
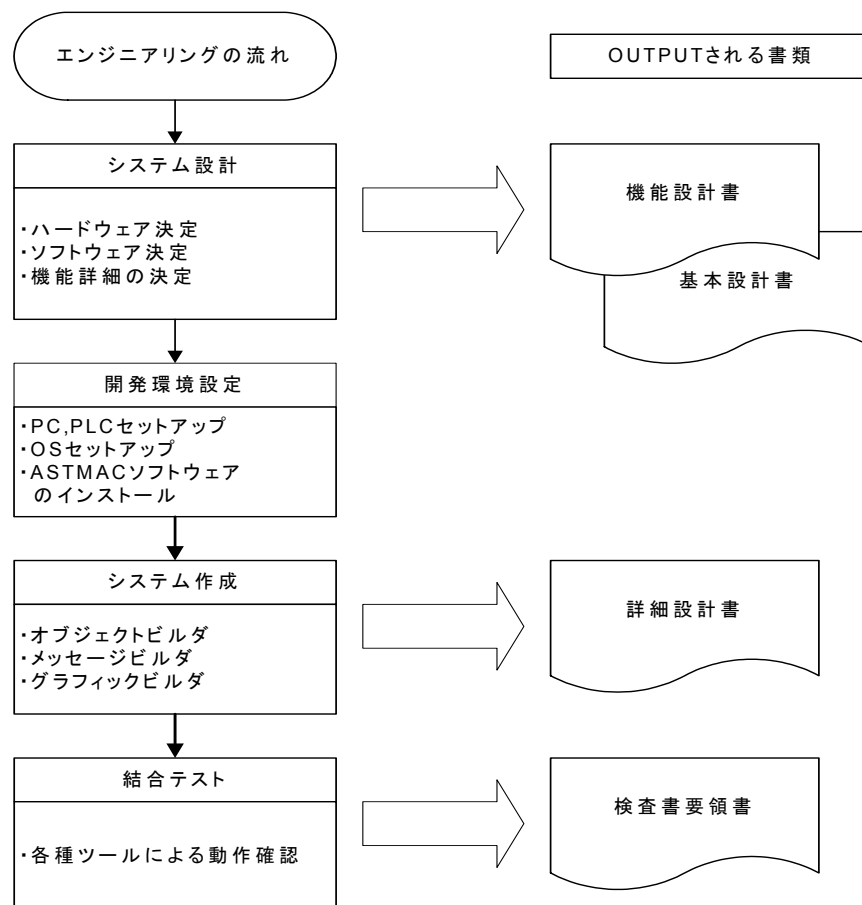
図　0.1 ASTMAC システム作成の流れ

このような流れでエンジニアリングを行うわけですが，本書では「図 0.1 ASTMAC システム作成の流れ」で示した「システム設計」について説明します。「システム設計」を ASTMAC 概説，機能設計，基本設計の 3 つの章に分けて説明します。

■ ASTMAC システム例

本書内では ASTMAC システムの例を統一的に用いています。
概要を「図 0.2 ASTMAC による自動車生産ラインの例」を用いて記述しています。
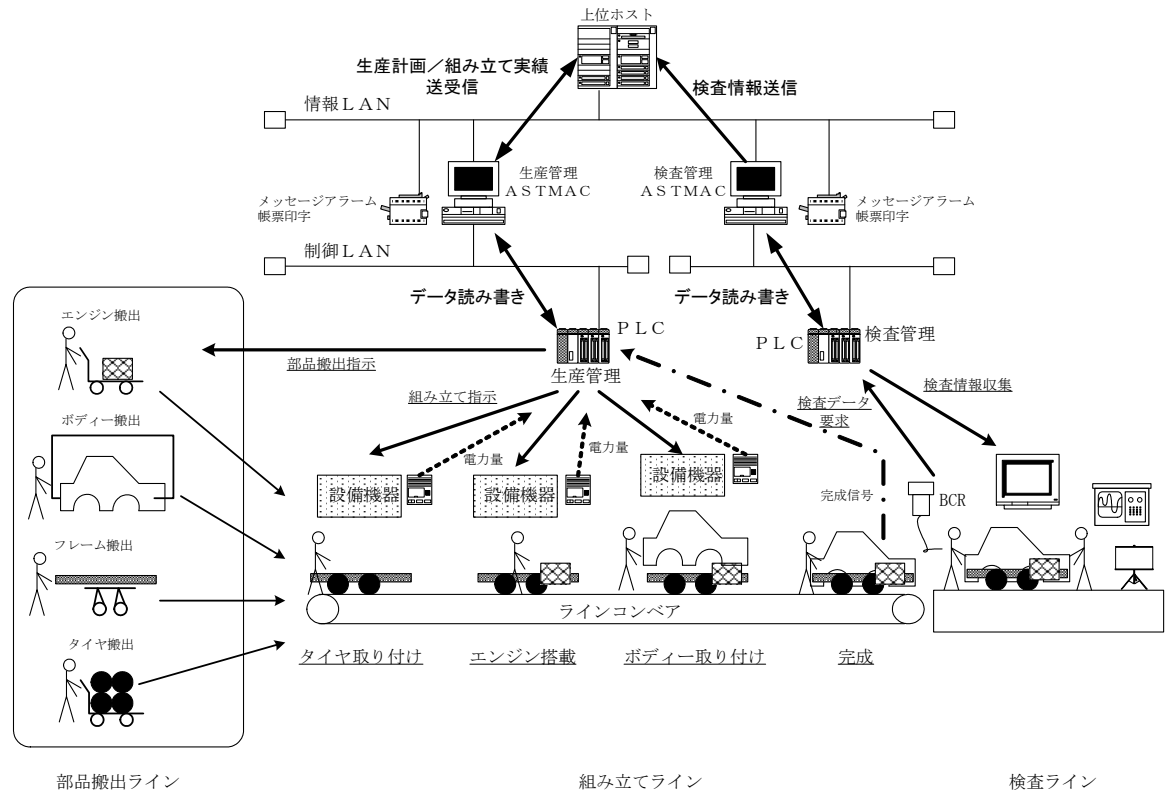
図　0.2 ASTMAC による自動車生産ラインの例

上記システムは自動車生産ラインの部品搬出ラインから組み立てライン，検査ラインまでを ASTMAC 2 台で管理することを想定しています。
2 台の ASTMAC をそれぞれ生産管理機と検査管理機として，各 ASTMAC は以下の機能を持つように設計します。

● 生産管理 ASTMAC（生産管理機能）

・ 上位ホストより生産計画データを一括受信する。
・ 表示画面は本日生産計画画面，組立ラインモニタ，組立実績一覧画面，アラームメッセージ履歴画面，消費電力モニタとする。
・ ラインコンベアと同期しながら部品搬出ライン，組み立てラインに対し搬出指示（ランプ点灯による指示），組み立て指示（設備機器にアナログ信号による指示）等の生産指示を行う。
・ 組み立て作業中の車両情報をモニタ表示する。
・ ライン稼動状況を監視し，ライン停止等の異常をアラームメッセージ表示／プリンタ出力する。
・ 組み立てラインの各工程の消費電力を常時監視し，トレンドグラフ表示する。
・ 組み立て完成した車両データ一覧を画面表示し，完成実績データとして保存する。
・ 日報処理により組立完成実績の帳票印字を行い，日報データを上位ホストへ一括送信する。

● 検査管理 ASTMAC（検査管理機能）

・ 検査対象車両データはバーコードリーダーで読み込まれる車両識別バーコード情報を基に組立実績データ（組立完了車両）を検索し，該当車両データを生産管理 ASTMAC から検査管理 ASTMAC へダウンロードする。
・ 表示画面は検査作業中の車両情報を表示する検査ラインモニタとアラームメッセージを表示するアラームメッセージ履歴画面とする。
・ 日報処理により検査情報を帳票印字及び上位ホストへ一括送信する。検査ラインからの検査情報（検査機器からのアナログ信号）を収集し，車両毎に検査情報ファイルに保存する。
・検査ラインの検査異常内容をプリンタに出力し，異常履歴ファイルに保存する。

商標

■ 商標ならびにライセンスソフトウェアについて

- STARDOM は，横河電機株式会社の商標です。
- その他，本文中に使われている会社名・商品名は，各社の登録商標または商標です。
- 本文中の各社の登録商標または商標には，TM，®マークは表示しておりません。

ASTMAC エンジニアリングガイド

TI 34P02K02-01 2 版

目次

1. ASTMAC概説

1.1 ASTMACとは

ASTMAC でのエンジニアリングを行う前に，ASTMAC とは何かを説明します。

ASTMAC は，“企業の 21 世紀成長を見通した生産ラインはどうあるべきか”，この長期的視野に立ち，業界標準の最新技術と横河電機が培ってきた生産ラインのノウハウを結集し，開発された生産ライン構築ソフトウェアです。
これまでの組み込み系の制御システムに比べ，汎用性と可搬性に優れており，最先端の生産ライン実現へ，エンジニアリング環境と操業環境の両面から最適ソリューションを提供します。
次の図のように，ASTMAC は ETS（Enterprise Technology Solutions）の Control の部分に位置します。
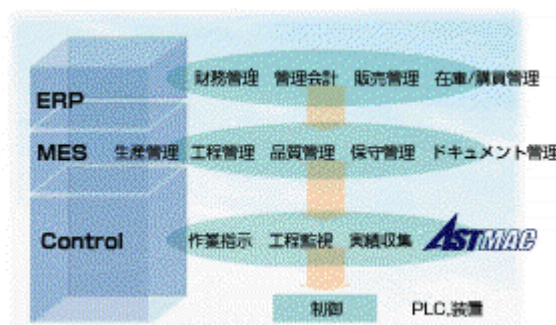
図　1.1 ETS コンセプトにおける ASTMAC の位置付け

1.2 ASTMACの特徴

次に ASTMAC の主な特徴を示します。
- 先進のオープン性
- 柔軟なシステム構成
- 高機能なオブジェクト指向
- イージーエンジニアリング
- 強力な操業支援機能

● 先進のオープン性

ASTMAC は閉ざされていた生産ラインを解放するため，オープンなプラットフォームに業界標準のコンポーネントやインタフェースなど先進のオープン技術を全面に採用しています。主に次のようなものが上げられます。
- PC-AT 互換機および汎用 PLC を採用
- Microsoft Windows XP,Windows 7 を採用
- Ethernet，RS-232-C などの標準インタフェースを採用
- Microsoft ActiveX アーキテクチャを採用
- VBA（Visual Basic Applications Edition）を全面採用
- OPC（OLE for Process Control）に対応

● 柔軟なシステム構成

ハードウェア，ソフトウェアとも小規模システムから大規模システムまで自由に構成することができ，プランに応じて拡張が可能です。

［ハードウェア構成］
マスタステーション 1 台，PLC1 台で構成する小規模システムから，マスタステーション 4 台，PLC32 台で構成する大規模システムまで，システムの規模に応じて自由に構成することができます。

［ソフトウェア構成］
システムの規模に応じて最適サイズの基本モデルを選択することができます。大規模システムの場合にも，ランタイム版の使用によりコストミニマムなシステムを構築できます。

● 高機能なオブジェクト指向

ASTMAC は横河電機のシステム技術・ノウハウを最新のオープン・オブジェクト指向テクノロジーと組み合わせています。
グラフィックは，マウス操作で図形を描画したり，部品を画面に貼り付けて設定するだけの簡単な操作で作成することができます。
また，生産管理や生産制御で要求される，高速検索処理やマルチタスク処理を簡単に実現するコンポーネントが用意されています。

● イージーエンジニアリング

ASTMAC は各種支援ビルダが用意されており，エンジニアリングの多くの場面でプログラムレスが図られております。

● 強力な操業支援機能

生産ライン制御システムの操業に不可欠な支援として，次の 4 つの機能が用意されています。

- メッセージ管理機能
- ヒストリデータ収集／トレンドグラフ表示機能（オプション）
- レポート管理／帳票出力機能（オプション）
- プロセス管理機能

［メッセージ管理機能］

ASTMAC のデータサーバから出されるアラームメッセージや情報メッセージ以外にアプリケーションとしてユーザが定義してアラームやメッセージを出すことも可能です。

［ヒストリデータ収集／トレンドグラフ表示機能］

収集周期，保存期間などを指定することにより，簡単にヒストリデータを収集することができます。また，グラフの定義等を行うだけでそのデータをもとにトレンドグラフを描画することができます。

［レポート管理／帳票出力機能］

定義した期間，タイミングで，現場の各種データをデータファイルへ保存し，Microsoft-Excel の帳票で出力します。

［プロセス管理機能］

システムプログラムやアプリケーションプログラムを統括管理し，登録・設定されたプログラムの起動やシャットダウンを自動的に行います。

このような ASTMAC の特徴を生かしたシステムとして，適用アプリケーションには，次のようなものがあります。

- 監視システム
- 連続プロセスの監視
- ＦＡ生産ラインの監視
- 作業指示システム
- 設備制御
- 物流管理
- 生産管理
- その他
  - 在庫管理
  - 検査システム

1.3　ASTMACの構成

今までの説明で ASTMAC の特徴などを知って頂けたと思いますが，では，基本的に ASTMAC はどのように構成されているかを説明します。
ASTMAC は，次の 3 つの機能で構成されています。

・ アプリケーションフォーム
・ データサーバ
・ I/O オブジェクト
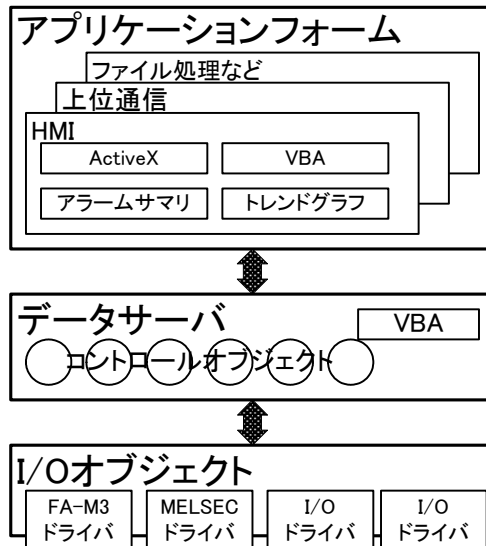
図　1.2 ASTMAC の構成

● アプリケーションフォーム

アプリケーションフォームでは，操作監視画面とアプリケーションプログラムが実行できます。高機能 ActiveＸコントロールを豊富にサポートし，ドラッグ＆ドロップの簡単操作で画面を作成できます。
アプリケーション作成は，VBA で行います。
アプリケーションフォーム間のデータ共有やイベント通知を容易に実現しました。（マルチタスク支援オブジェクト）

● データサーバ

データサーバは ASTMAC システムの中心に位置し，I/O，ヒューマンインタフェースとのデータ交換を行うための ActiveＸ対応コンテナです。イベント駆動型の強力なデータ収集エンジンでリアルタイム性を追求しました。
VBA エンジン搭載により，オブジェクトの機能拡張やデータ処理が実現可能です。
ＯＰＣオートメーションインタフェース（*1）を装備しています。

*1： OPC-F（OLE for Process Control Foundation）が標準化を進めている Visual Basic や他の VBA 等からのアクセスが可能なインタフェースです。OPC DA 2.0 をサポートしています。

● I/O オブジェクト

I/O オブジェクトはフィールド機器と通信を行います。
機器ごとの固有なインタフェースを隠蔽し，データサーバから共通の手順でアクセスします。

2. 機能設計

2.1 機能設計の流れ

本章では，ASTMAC を使用したシステムの機能設計を行うための指針をシステム例を用いいながら示します。
以下に機能設計の流れを示します。
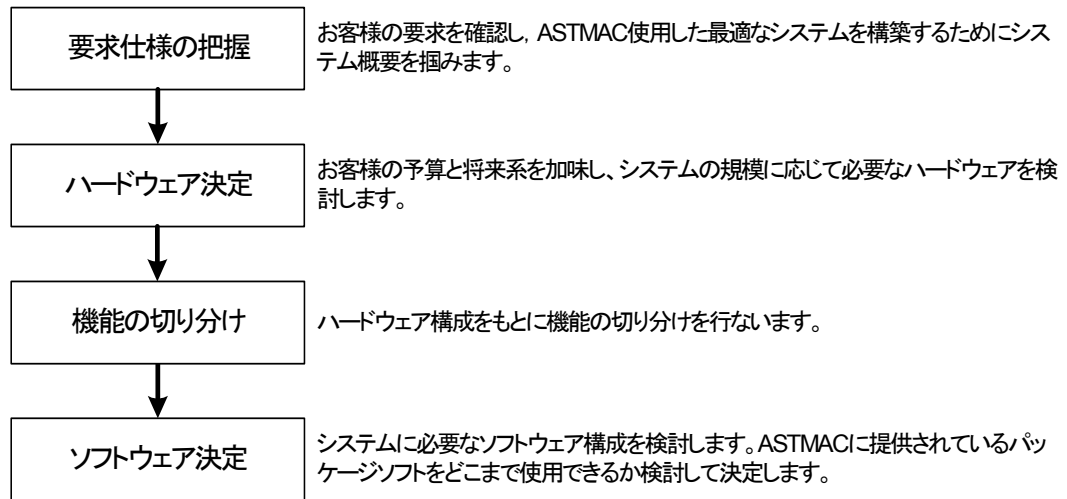
図　2.1 機能設計の流れ

2.2　要求仕様の把握

お客様の要求を確認し，“何をどうしたいのか”，“何をどうするシステムなのか”概要を掴み，ASTMAC を使用できるシステムなのか見極めます。
主に次の内容について，仕様を決定していくことにより ASTMAC のシステム構築を行うことができます。

・ I/O 仕様
・ 機能項目
・ データ保存項目

● I/O 仕様

次の表を使用して，I/O の点数を決定します。
自動車生産ラインの管理を行うシステムの例を参考にして，次のような表を作成します。

表　2.1 I/O 表

No	項目名称	入力点数	出力点数
1	部品搬出指示		50
2	組立ライン電力量	30	
3	組立ライン生産指示		30
4	完成信号	10	
5	検査情報	100	100
	計	140	180

● 機能項目

詳細の画面，帳票のレイアウト，表示項目などについては基本設計の方で決定しますので，ここでは，どんな機能が必要で，画面として表示させるのか帳票として印刷させるのかを決定します。
自動車生産ラインの管理を行うシステムの例を参考にして，次のような表を作成します。

表　2.2 機能項目

	機能項目	処理内容	画面の有無	帳票の有無
1	生産計画受信	上位ホストから生産計画データを受信し，ＨＤに保存する。		
2	生産指示	生産計画データをもとに部品搬出ラインと組立ラインに指示を出す。	○	
3	消費電力管理	組立ラインの電力量を取り込み，グラフとして表示する。	○	
4	生産実績管理	組立ラインの完成信号を取り込み，生産実績を管理する。	○	
5	生産日報処理	生産実績を日報処理して，帳票として印字し，上位ホストへ送信する。	○	○
6	稼働状況監視	組立ラインの稼働状況を監視し，異常が発生した場合，画面表示，プリンタ印字を行う。	○	○
7	検査指示	バーコードリーダーで車両識別情報を読込，それをベースに検査指示を検査ラインに出す。	○	
8	検査情報管理	検査ラインから検査情報を取り込み，ＨＤに保存する。		
9	検査日報処理	検査情報を日報処理して，帳票として印字し，上位ホストへ送信する。	○	○
10	検査異常監視	検査異常が発生した場合，画面表示，プリンタ印字を行う。	○	○

● データ保存項目

詳細のファイル仕様については基本設計の方で決定しますので，ここでは，保存するデータとしてどんな項目をどのぐらい保存するのかを決定します。
自動車生産ラインの管理を行うシステムの例を参考にしながら，次の表に記述します。

表　2.3 データ保存項目

No	保存データ名	保存内容	データ数 または保存期間
1	生産計画	1 日の生産計画数を保存する。	1 日
2	組立実績	組立完成した車両データを保存する。	1 年
3	電力データ	組立ラインの消費電力データをグラフ表示用として保存する。	1 月
4	検査情報	検査ラインで検査した結果の検査情報を保存する。	1 年

2.3 ハードウェア決定

2.3.1 ハードウェア選定の指針

要求仕様は，把握できましたか？
では，次にその要求仕様を実現するためのハードウェアを決定します。
ASTMAC を使用したシステム構成では，次の点に留意して決定する必要があります。

・ マスタステーションの台数
・ クライアントステーションの台数
・ コントローラの台数
・ LAN 構成
・ プリンタの接続

● マスタステーションの台数

マスタステーションとは，コントローラのデータを取り込むもので，ASTMAC システムの中核に位置します。
1 システムで，最低 1 台は必要です。同一制御 LAN 上には最大 4 台（*1）まで接続できます。

*1： 台数が仕様を超える場合は，弊社までご相談ください。

● クライアントステーションの台数

クライアントステーションは，マスタステーションで管理している情報の監視モニタやアラーム表示などを行うことが可能です。
1 マスタステーションに対し，最大 16 台（*1）まで接続可能です。

*1： 台数が仕様を超える場合は，弊社までご相談ください。

● コントローラの台数

標準コントローラとして，横河電機製 FA-M3 を使用します。FA-M3 以外の PLC を使用する場合は，オプションの I/O ドライバにより接続することができます。また，I/O ドライバが提供されていない特殊な装置などは，ユーザが I/O ドライバを作成することにより，接続することができます。
コントローラは，異機種混合が可能で，合計 32 台（*1）まで接続することができます。

*1： 台数が仕様を超える場合は，弊社までご相談ください。

● LAN 構成

LAN 構成は，情報系 LAN と制御系 LAN の 2 つに大別されます。この 2 つの LAN は，分離するように構成します。

［情報系 LAN］

情報系 LAN とは，上位情報系サーバマシンやクライアントステーションを接続する LAN です。業界標準の Ethernet を使用します。

［制御系 LAN］

制御系 LAN とは，マスタステーションとコントローラを接続する LAN です。業界標準の Ethernet を標準としています。システムの規模が小さい場合は，LAN 接続でなく，RS-232C で接続することも可能です。

● プリンタの接続

プリンタは，情報系 LAN 上にネットワークプリンタとして接続します。接続には，業界標準の Ethernet を使用します。情報系 LAN 上に接続しないで，ASTMAC がインストールされているパソコンからパラレルポート他で接続することも可能です。

2.3.2 ASTMAC一般的なハードウェア構成図

ASTMAC の一般的なハードウェア構成図を規模に応じて次に示します。

● 小規模

マスタステーション 1 台，PLC1 台，プリンタ 1 台（図 2.2 小規模構成図）
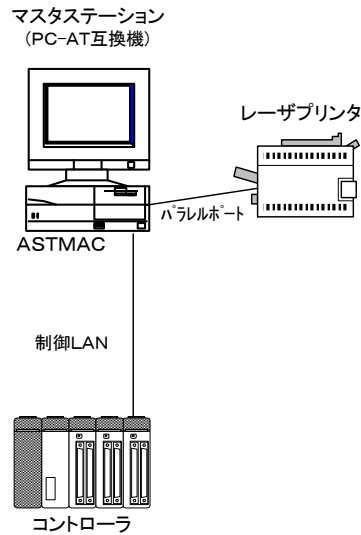
図 2.2 小規模構成図

● 中規模

マスタステーション 1 台，PLC4 台，プリンタ 1 台，クライアントステーション 1 台（図 2.3 中規模構成図）
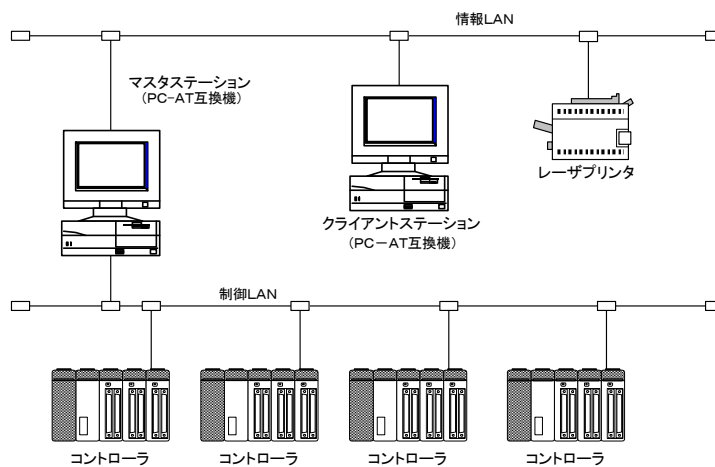
図 2.3 中規模構成図

● 大規模

マスタステーション 2 台，PLC4 台，プリンタ 2 台，クライアントステーション 2 台，上位サーバ 1 台（図 2.4 大規模構成図）
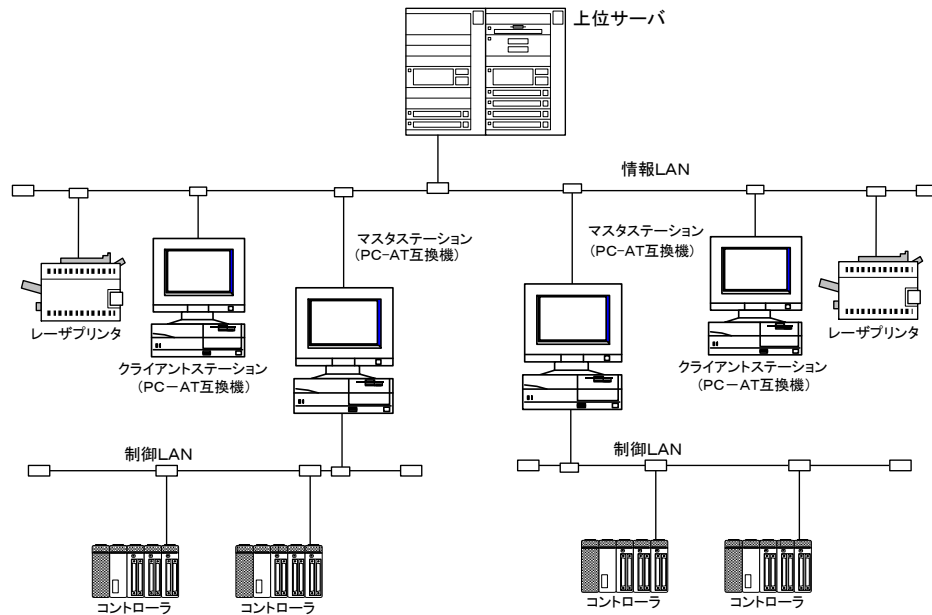
図　2.4 大規模構成図

2.3.3 ハードウェア構成図の例

中規模構成図の例として，自動車生産ラインの管理を行うシステムのシステム構成図を，以下に示します。
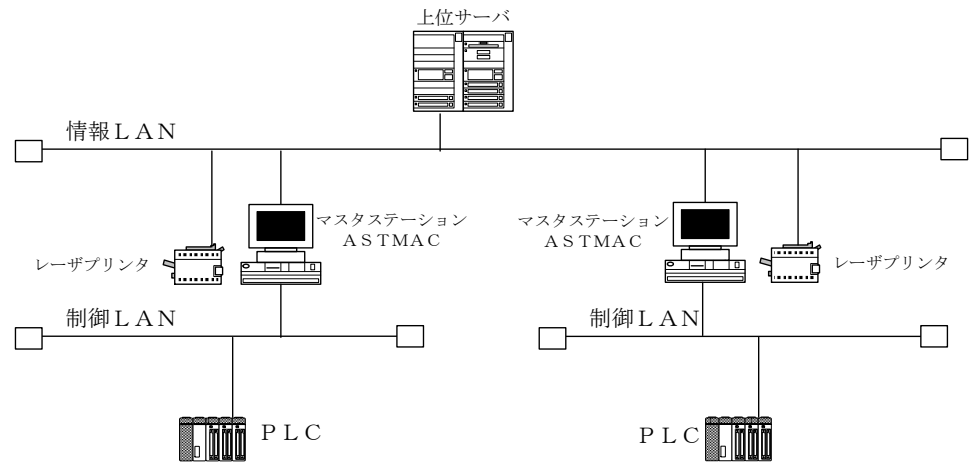
図　2.5 自動車生産ラインの管理システム

ハードウェアの構成が決定できましたので，具体的な機器を決定します。推奨機器については，”ASTMAC 友の会”のホームページをご覧ください。
自動車生産ラインの管理を行うシステムの例を参考にして，次のような表を作成します。

表　2.4 ASTMAC ハードウェア構成表

No	機器名	機器	製品名	メーカー	主な仕様	台数	備考
1	生産管理機	PC	Windows7	H社	1 GHz 以上の 32 bit (x86) プロセッサ または 64 bit (x64) プロセッサ Memory　2GB HDD　20GB CRT　1024 × 768 ドット	1	
2	生産管理プリンタ	LBP-730		K社		1	
3	生産管理コントローラ	FA-M3		横河電機	BASIC ﾓｼﾞｭｰﾙ Ethernet ﾓｼﾞｭｰﾙ RS422 ﾓｼﾞｭｰﾙ 接点入力　8 点 接点出力　8 点 ｱﾅﾛｸﾞ出力　4 点	1	
4	検査管理機	PC	Windows7	H社	1 GHz 以上の 32 bit (x86) プロセッサまたは 64 bit (x64) プロセッサ Memory　2GB HDD　20GB CRT　1024 × 768 ドット	1	
5	検査管理プリンタ	LBP-730		K社		1	
6	検査管理コントローラ	FA-M3		横河電機	BASIC ﾓｼﾞｭｰﾙ Ethernet ﾓｼﾞｭｰﾙ RS422 ﾓｼﾞｭｰﾙ ｱﾅﾛｸﾞ入力　4 点	1	
7	ハブ				8ﾎﾟｰﾄ	2	
8	Ethernet ｹｰﾌﾞﾙ	100BASE-T			15m	2	
9	Ethernet ｹｰﾌﾞﾙ	100BASE-T			30m	4	

2.4 機能の切り分け

システムの仕様およびハードウェアが決定しました。では次にどの機器でどのような機能を行わせるのか，機能の切り分けを行います。

ASTMAC を利用したシステムでは，基本的に上位コンピュータ，ASTMAC，PLC という形に分割され，ASMTAC は上位コンピュータと PLC の間に位置し，それぞれと通信することでシステムを実現します。
上位コンピュータ，ASTMAC，PLC のそれぞれの役割は次のように分担します。

● 上位コンピュータ

・ DB 管理
・ 生産計画，実績管理など

● ASTMAC

・ ヒューマンマシンインタフェース
・ グラフィック監視画面，設定，操作画面，警報表示，グラフ表示など
・ データの収集，加工，処理
・ 収集データの加工，保存，判断，指示，警報処理，トレンドデータ記録，レポート処理など
・ 生産管理
・ 生産情報の加工，指示，実績データの収集，加工，保存
・ 上位通信
・ ADO によるデータベースアクセスなど

● PLC

・ 装置制御
・ フィールド信号の収集，変換
・ フィールド機器との通信
・ 高速データ処理
数 100ms 以下の応答を求められるデータ処理
・ ネットワークやＰＣから切り離されても基本的な制御を継続し，最低限のデータを保存する処理

2.4.1 機能の切り分け例

自動車生産ライン管理のシステムの機能の切り分けを行うと次のようになります。
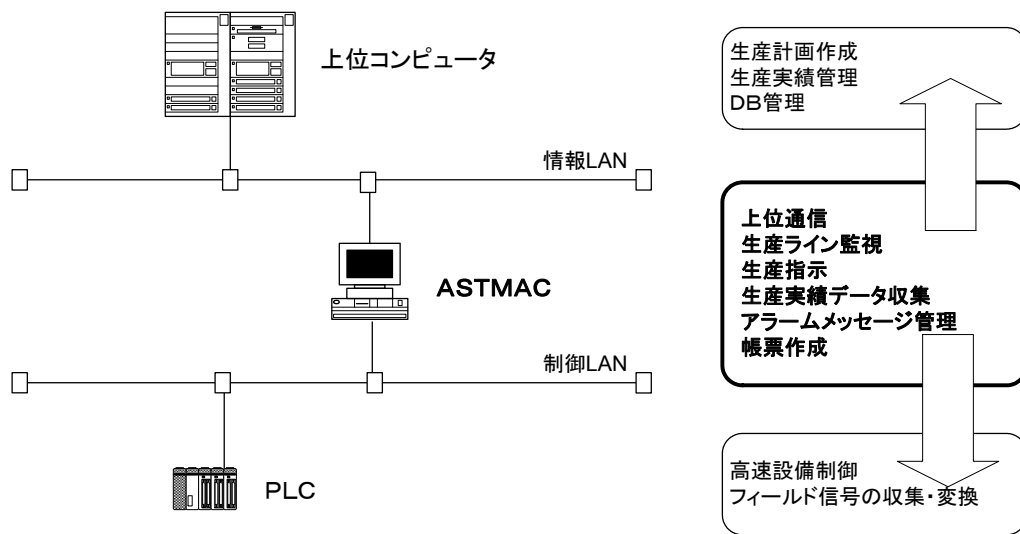
図 2.6 上位コンピュータ，ASTMAC，PLC の役割分担

上記システムにおいて ASTMAC は以下のような役割を持ちます。

1. 上位コンピュータとデータの送受信を行う。
2. 生産ラインへ生産指示を行う。
3. 生産ラインの運転状態の監視を行う。
4. 生産実績を収集する。
5. 生産ラインからのフィールド情報を収集する。
6. 帳票を作成する。
7. アラームメッセージ管理を行う。

これらの役割を果たすために ASTMAC は上位コンピュータや PLC との情報のやり取りを行う必要があります。
ASTMAC が上位コンピュータとの情報のやり取りをすることを上位通信とし，PLC との情報のやり取りをすることを PLC 通信とします。

上記の説明では，上位通信，PLC 通信という観点で説明しましたが，更に，大規模システムになりますと，複数の ASTMAC を使用する場合が考えられます。その場合は，それぞれの ASTMAC での機能の切り分けを行う必要があります。
ASTMAC ではマスタステーション，クライアントステーションなどにより機能が異なりますので，それを意識して機能の切り分けを行います。

特に次のことに気を付けてください。

- 1 台のコントローラのデバイスを複数の ASTMAC から重複してアスセスする設計はデータ収集のパフォーマンス低下以外に，オブジェクトの登録・デバッグ・保守の作業効率も低下させることになるので，行わないようにしてください。
- 1 つの画面の中に，4 台の ASTMAC データを表示させることができます。
- 1 つの ASTMAC データを，クライアントステーションの画面を含めて同時に最大 16 画面で表示することができます。

上記のことを考慮して，自動車生産ラインの管理を行うシステムの例を参考にして，次のような表を作成します。

表 2.5 自動車生産ライン管理 ASTMAC の機能

No	機器名	I/O 項目	保存項目	機能項目	画面名称	帳票名称
1	生産管理機 （ASTMAC） （ﾏｽﾀｽﾃｰｼｮﾝ）		生産計画 組立実績 電力データ	生産計画受信 生産指示 消費電力管理 生産実績管理 生産日報処理 稼働状況監視	生産計画 消費電力モニタ アラームメッセージ	生産日報 生産管理アラーム
2	生産管理プリンタ					生産日報 生産管理アラーム
3	生産管理 PLC	部品搬出指示 組立ライン電力量 組立ライン生産指示 完成信号				
4	検査管理機 （ASTMAC） （ﾏｽﾀｽﾃｰｼｮﾝ）		検査情報	検査指示 検査情報管理 検査日報処理 検査異常監視	検査指示 検査情報 検査異常メッセージ	検査日報 検査異常
5	検査管理プリンタ					検査日報 検査異常
6	検査管理 PLC	検査情報				

システム構成図をもとに各機器に上の表でまとめた機能を記述していくと，「図 2.7 システム概念図」が作成されます。
実際の運用を絡め，ハードウェア構成にデータの流れを盛り込んだ図を作成します。
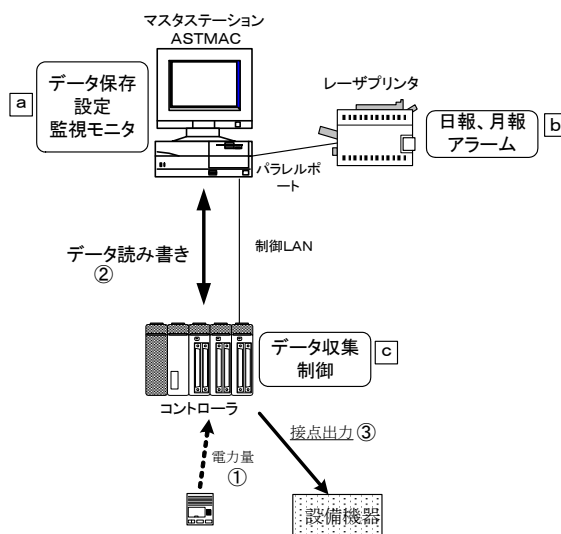
図 2.7 システム概念図

自動車生産ラインの管理を行う ASTMAC システムを例として以下に示します。
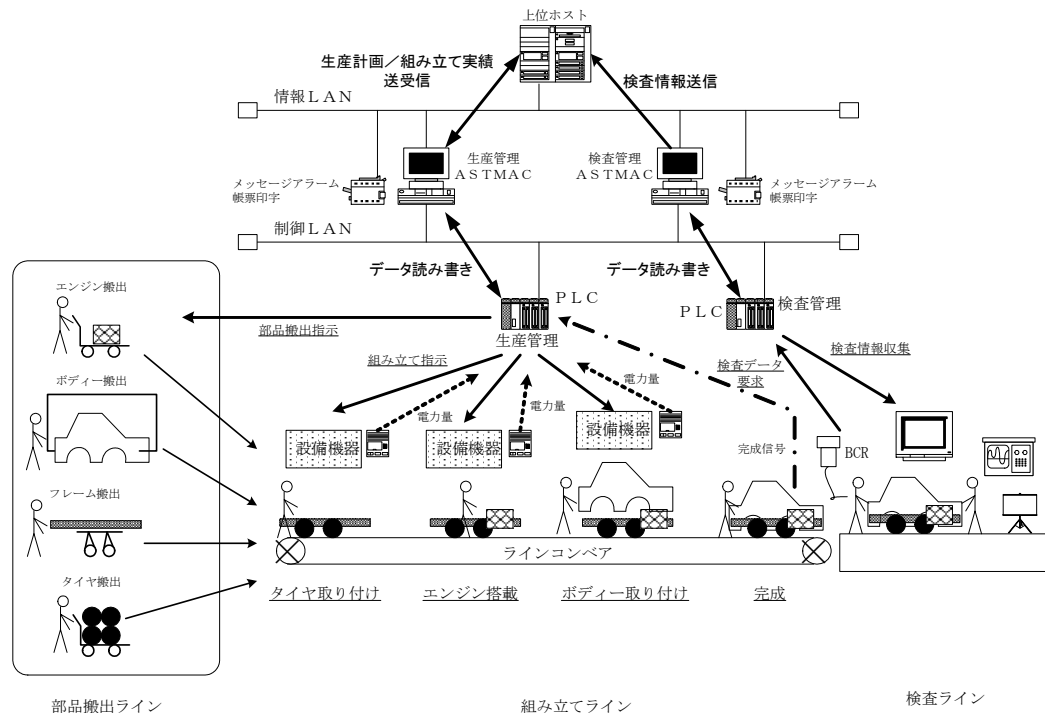
図　2.8 システム概念図（自動車生産ライン）

「図 2.8 システム概念図（自動車生産ライン）」は自動車生産ラインの部品搬出ラインから組み立てライン，検査ラインまでを ASTMAC2 台で管理しています。2 台の ASTMAC をそれぞれ生産管理機と検査管理機として構築しています。生産管理の ASTMAC，コントローラは，部品搬出ラインと組立ラインのデータ収集，制御を行い，検査管理の ASTMAC，コントローラは検査ラインのデータ収集を行います。両方の ASTMAC とも，実績データを上位ホストに転送します。

2.5　ソフトウェア決定

2.5.1　基本モデルの選択

さて，機能も機器も決定しましたので，このシステムを実際に実現するために必要となるソフトウェアを決定します。
ASTMAC には基本モデルとオプションパッケージがありますが，初めに基本モデルの検討を行います。

ASTMAC の基本モデルには，次の 2 種類があります。

- ベースタイプ
- ビュークライアントタイプ

表　2.6 ASTMAC 基本モデルの選択

基本モデル	説明		データサーバ仕様		
			実行オブジェクト数	接続 I/O 機器数	ｸﾗｲｱﾝﾄ接続数
ベースタイプ	標準的なモデルで，ほぼ全てのアプリケーションに適用できます。 フルタイム版とランタイム版のライセンスがあります。	特大規模：LL	～ 5000	32	16
		大規模：L	～ 2000	32	16
		中規模：M	～ 1000	32	16
		小規模：S	～ 200	32	16
ビュークライアントタイプ	ＨＭＩアプリケーションを動作させるモデル。 情報系ＬＡＮを通じ，リモートでマスタステーションと接続します。 リモートオペレーション端末用です。		－	－	－

上の表を参考に基本モデルを選択してください。
パソコン 1 台につき，基本モデルのいずれかが必要です。

● ベースタイプ

システム開発を行う場合は，フルタイム版が必須です。違うパソコンで全く同じような機能を行う場合は，フルタイム版とランタイム版の組み合わせでも結構ですが，機能として別のことを実現する場合は，機能の違うパソコンごとにフルタイム版を選択してください。
例として用いている自動車生産ラインのシステムでは，2 台のパソコンが生産管理と検査管理という，それぞれ違う機能を行いますから，両方にベースタイプのフルタイム版が必要になります。

Ｓ，M，Ｌ，LL の選択は，オブジェクト数により決定しますので，オブジェクト数を算出してください。次の方法で概略算出することができます。

オブジェクト数 = タスク数＋ I/O 数 + 20

［タスク数］
次の機能を 1 つとしてカウントします。（最大 16 タスク）
・ シグナルオブジェクトの個数

［I/O 数］
次のようにカウントします。
・ アナログデータは 1 カウント
・ 割り込み入力は 1 カウント
・ ステータス（ビットの塊データ）は 1 カウント
・ バーコード入力は 1 カウント
・ ブロックデータとテキストは 128 バイトごとに 1 カウント

［20］
予備として加算します。

● ビュークライアントタイプ

マスタステーションの機能を別の場所で操作監視するような用途の場合に選択します。
データサーバ機能が不要で，操作監視画面，アプリケーション処理のみ必要な場合はビュークライアントタイプを選択します。

2.5.2 オプションパッケージの選択

基本モデルが決定できましたら，次にオプションパッケージの検討を行います。
ASTMAC には，次のオプションパッケージが用意されています。必要に応じて追加してください。

表 2.7 ASTMAC オプションパッケージ

基本モデル / オプションパッケージ	ベースタイプ	ビュークライアントタイプ
トレンドパッケージ	○	○
帳票パッケージ	○	
テスト機能パッケージ	○	
ＭＥＬＳＥＣ接続パッケージ	○	
ＹＥWMＡＣ接続パッケージ	○	
マルチタスク支援パッケージ	○	○
ＳＹＳＭＡＣ接続パッケージ	○	
OPC サーバ接続パッケージ	○	
FA-M3 計装パッケージ	○	
ＤＡＲＷＩＮ接続パッケージ	○	
カスタムドライバ接続パッケージ	○	○注
データ入力コントロール	○	○
表形式表示コントロール	○	○
グラフ表示コントロール	○	○

注： 動作環境仕様，サポート機能など，一部変更あり

● トレンドパッケージ

ヒストリデータの変化を監視するための機能で，定周期でコントロールからデータを収集し，値の変化を画面上にグラフ（トレンドグラフ）表示する機能を提供します。機能の詳細については，基本設計を確認してください。

● 帳票パッケージ

帳票パッケージは，Excel シートを使用して簡単に帳票を作成することができます。帳票パッケージを動作させる場合は，「Microsoft 日本語 Excel2007/Excel2010」を追加する必要があります。

● テスト機能パッケージ

テスト機能パッケージには，次の 3 つの機能があり，有効なデバッグ環境を提供するものです。基本モデルのベースタイプ，フルタイム版をインストールするパソコンには，追加してください。

- デバイスモニタ機能
- オブジェクトトレース機能
- シミュレート機能

機能の詳細はデバッグの項を参照してください。

● MELSEC 接続パッケージ

MELSEC 接続パッケージは，三菱電機製シーケンサ MELSEC 用の I/O ドライバをサポートするものです。MELSEC を使用する場合は，追加してください。

● YEWMAC 接続パッケージ

YEWMAC 接続パッケージは，弊社製 FA コンピュータ YEWMAC500 用の I/O ドライバをサポートするものです。YEWMAC500 を接続する場合は，追加してください。

● マルチタスク支援パッケージ

複数の実行プログラムを同時に実行させる場合，互いに協調しあいながら動作する必要があります。これを本書ではマルチタスク動作と呼びます。
ファイル処理と画面表示など複数の機能が共存する場合は追加します。
マスタステーションでは複数のタスクが動作しますので，必ず追加してください。

● SYSMAC 接続パッケージ

SYSMAC 接続パッケージは，オムロン社製 PLC の SYSMAC と接続するためのパッケージです。SYSMAC を使用する場合は，追加してください。

● OPC サーバ接続パッケージ

OPC サーバ接続パッケージは，I/O オブジェクトに OPC クライアントの機能を提供し，サードパーティ製の OPC サーバとのデータ交換を実現します。
OPC サーバ接続パッケージにより，ASTMAC と PLC やその他の機器メーカが提供する OPC サーバを接続する場合は追加してください。

● FA-M3 計装パッケージ

FA-M3 計装パッケージは，当社 FA-M3 をベースにフィードバック制御とシーケンス制御を実行する制御ユニットと，ASTMAC をインストールしたパソコンを制御 LAN で結び，コストパフォーマンスに優れたプロセス制御システムを構築するためのオプションパッケージです。

● DARWIN 接続パッケージ

DARWIN 接続パッケージは，ASTMAC システムに当社製 DARWIN（データアクィジションユニット）を接続するためのオプションソフトウェアです。また，本パッケージで，レコーダ DAQSTATION を接続することが可能です。DARWIN を使用する場合は，追加してください。

● カスタムドライバ接続パッケージ

Microsoft Visual Basic を使用して開発したカスタムドライバ（I/O ドライバ）を ASTMAC システムに組み込むためのパッケージです。
Windows7 では，動作環境仕様，サポート機能など，一部変更があります。

● GKitOCX ASTMAC Edition

GKitOCX ASTMAC Edition は，画面のアプリケーション構築に有益な ActiveX コントロール群の総称です。
コントロールの種類は，次の 3 種類が存在します。

・ NT310AJ データ入力コントロール
・ NT311AJ 表形式表示コントロール
・ NT312AJ グラフ表示コントロール

2.5.3 ソフトウェア決定の例

上記のようなことを考慮して検討した結果，ソフトウェア構成表を作成し，使用するソフトウェアを明確にします。
構成表として記述する項目を次に示します。

自動車生産ラインの管理を行うシステムの例を参考にて，次のような表を作成します。

表　2.8 ソフトウェア構成表（例）

No	機器名	種類	ソフト名	Ｖｅｒ	メーカー	備考
1	生産管理機 ASTMAC ﾏｽﾀｽﾃｰｼｮﾝ	ＯＳ	Windows7 Professional		Microsoft	
		帳票ソフト	Microsoft　Excel	Excel2010	Microsoft	
		ﾊﾟｯｹｰｼﾞ	ASTMAC ﾍﾞｰｽﾀｲﾌﾟ	R7.01	横河電機	
		ﾊﾟｯｹｰｼﾞ	ASTMAC トレンド表示	R7.01	横河電機	
		ﾊﾟｯｹｰｼﾞ	ASTMAC マルチタスク支援	R7.01	横河電機	
		ﾊﾟｯｹｰｼﾞ	ASTMAC 帳票機能	R7.01	横河電機	
		ﾊﾟｯｹｰｼﾞ	ASTMAC テスト機能	R7.01	横河電機	
2	検査管理機 ASTMAC ﾏｽﾀｽﾃｰｼｮﾝ	ＯＳ	Windows7 Professional		Microsoft	
		帳票ソフト	Microsoft　Excel	Excel2010	Microsoft	
		ﾊﾟｯｹｰｼﾞ	ASTMAC ﾍﾞｰｽﾀｲﾌﾟ	R7.01	横河電機	
		ﾊﾟｯｹｰｼﾞ	ASTMAC マルチタスク支援	R7.01	横河電機	
		ﾊﾟｯｹｰｼﾞ	ASTMAC 帳票機能	R7.01	横河電機	
		ﾊﾟｯｹｰｼﾞ	ASTMAC テスト機能	R7.01	横河電機	

3. 基本設計

3.1 基本設計の流れ

機能設計ではシステム構成図を作成しました。
次に基本設計ではシステムのインタフェースを明確にします。本章では，ASTMAC システムの基本設計を行うための指針を自動車生産システムの例を用いながら解説します。
まず始めに，基本設計の流れを以下に示します。
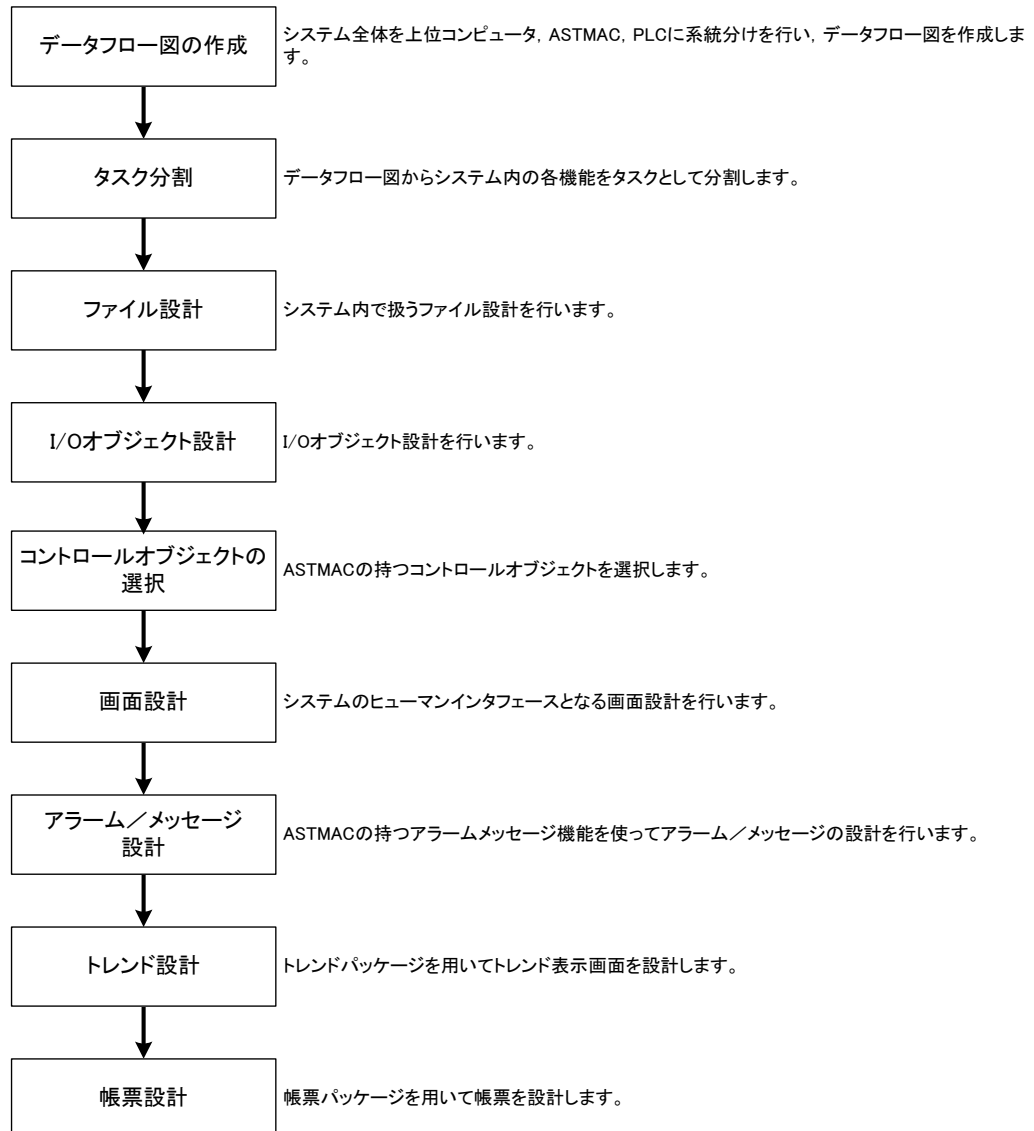
図 3.1.1 基本設計の流れ

3.2 データフロー図の作成

機能設計にて作成したシステム構成図を元に，データフロー図を作成します。
データフロー図は，機能の関連とデータの流れを図に示したものです。
データフロー図を作成することで，システム全体を体系的に整理することができ，ASTMAC システム内のタスク分割やファイル設計，画面設計を行うための指針となります。

ASTMAC システムでデータフロー図を考える上では
・ ASTMAC と上位コンピュータ
・ ASTMAC と PLC
・ ASTMAC と ASTMAC
・ ASTMAC 内の各タスク間
で受け渡されるデータを図として明確にする必要があります。

以下にデータフロー図を作成するまでの作業手順を示します。
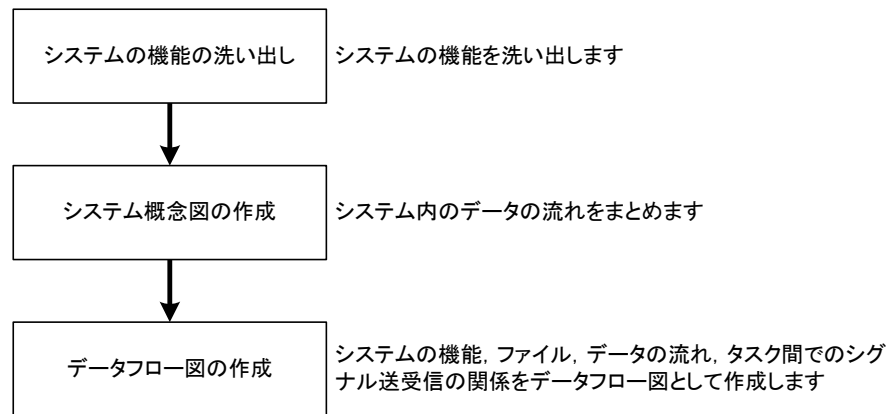
図　3.2.1 データフロー図作成の流れ

3.2.1 データフロー図の作成例

下記に，自動車生産ラインをモデルとした場合のシステム概念図と，この概念図をデータフローにした図を示します。
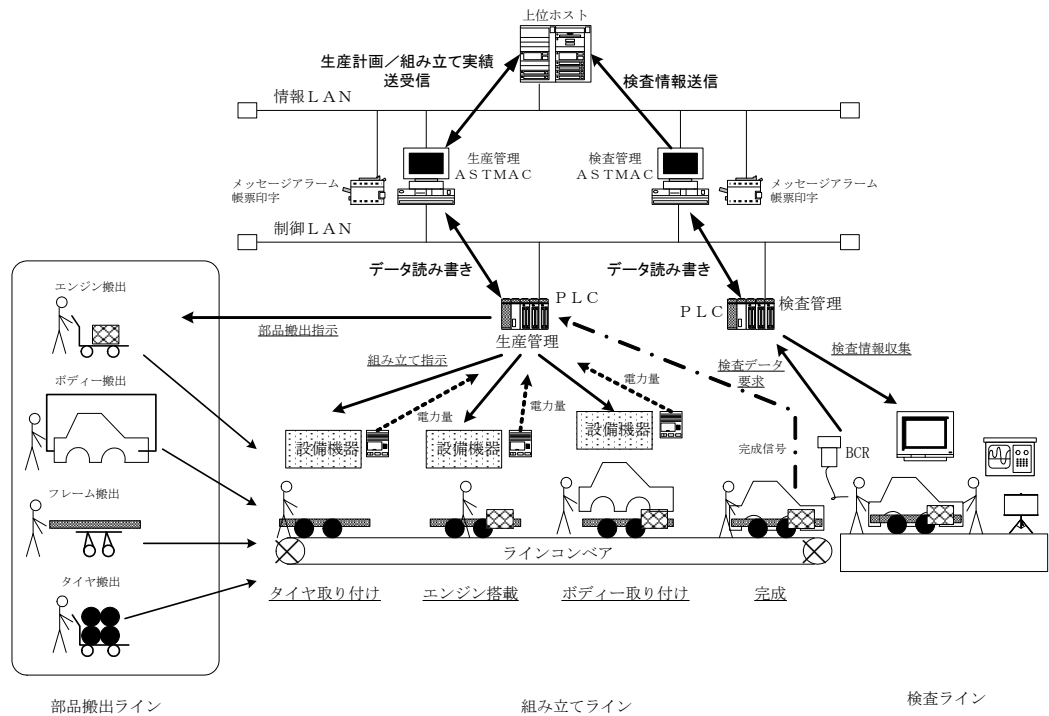
図　3.2.2 システム概念図

自動車生産ラインの「部品搬出＋組み立てライン」，「検査ライン」を ASTMAC で管理しています。

● 生産管理 ASTMAC の概要

部品搬出ラインと組み立てラインに生産指示を出し，かつ組み立て実績を収集し，帳票を作成します。また，組み立てラインの消費電力のトレンド表示も行います。

● 検査管理 ASTMAC の概要

検査ラインへ検査指示を出し,かつ検査情報を収集し，帳票を作成します。
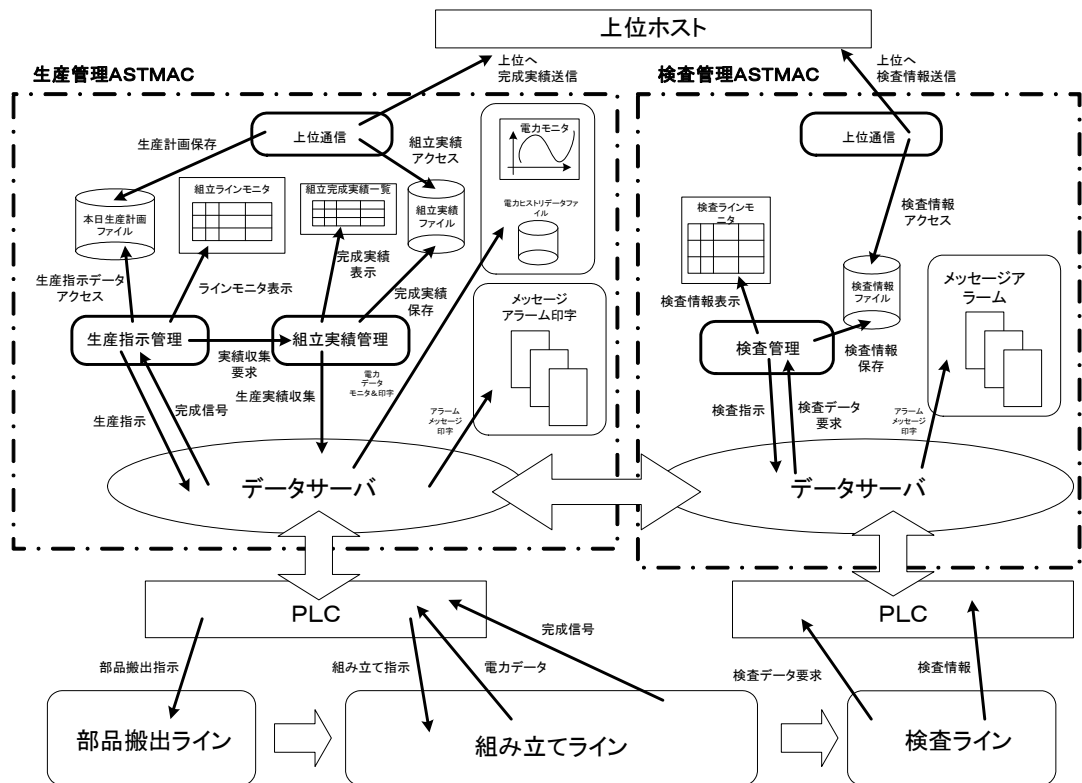
図　3.2.3 データフロー図

ASTMAC と上位コンピュータ，ASTMAC と PLC，ASTMAC と ASTMAC，ASTMAC 内のデータの流れ及び機能について明確にすることができました。

3.3 タスク分割

3.3.1 タスク分割の指針

データフロー図を作成したことにより，システム内の機能の位置づけが明確になりました。
通常のシステム内は上位通信，生産指示，実績管理など複数の機能が存在します。
ASTMAC システムではこれらの個々の機能を一つのタスクとすることを推奨します。
本書で述べるタスクとは，一つの実行モジュールの単位です。
ASTMAC では，グラフィックビルダで作成するアプリケーションフォーム（以下 AP フォームと略します）がタスクとなります。

機能＝タスク（AP フォーム）

注： ASTMAC では最大 16 タスク使用することができます。

一般に，タスク分割を行ったシステムは
- 多くの処理を効率的に並行処理することができます
- 特定処理による他処理の停止や待ち時間の発生が無くなります
- システムの機能拡張が容易です

等の利点があります。

3.3.2 タスク分割の例

では実際に，「図 3.2.3 データフロー図」で作成した図から自動車生産システムの生産管理 ASTMAC の機能をタスク分割すると以下のようになります。
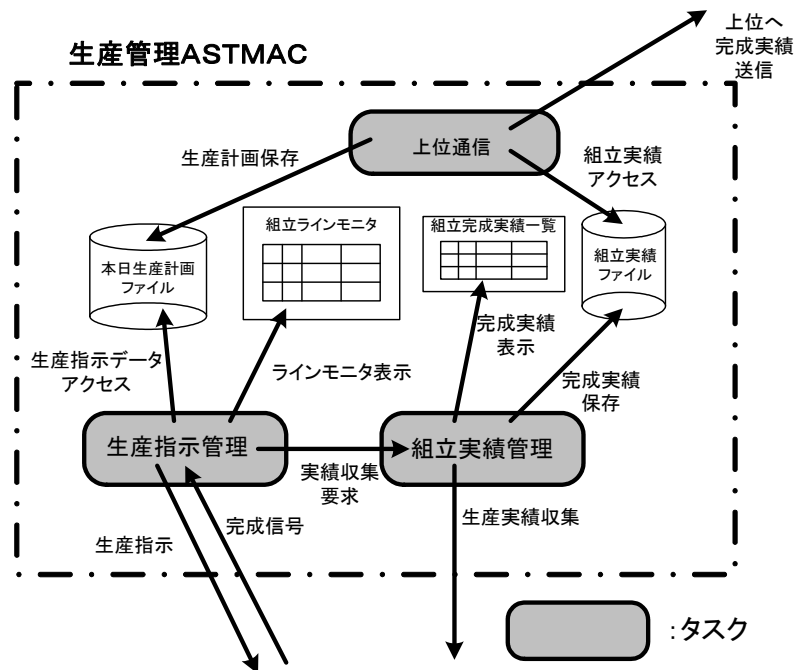
図 3.3.1 タスク分割の例

3.4 ファイル設計

システムを構築するにはデータを保持したり情報を検索するなど，ファイルやデータベース等が必要になります。
データベースを利用する場合は，上位コンピュータのデータベースサーバにアクセスが可能です。
本章では，ASTMAC システムのファイル設計についての基本方針を示します。
ファイル設計は一般に以下の手順で行われます。
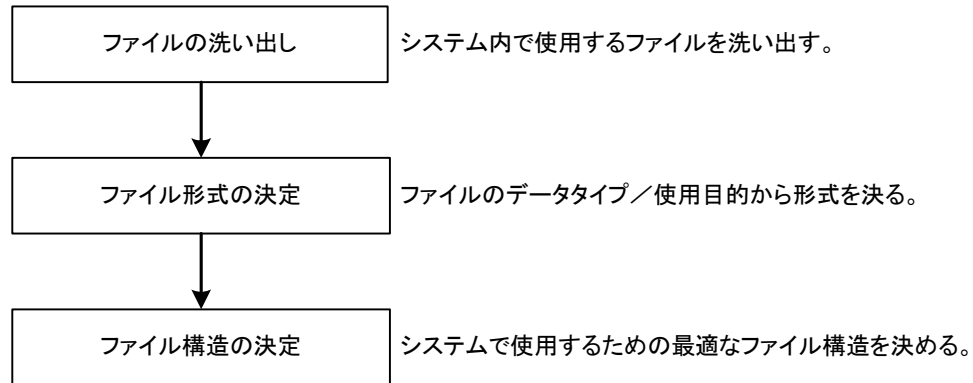
図 3.4.1 ファイル設計手順

3.4.1 ファイルの洗い出し

構築するシステムにおいて使用するファイルの検討を行います。システムとして必要なファイルをデータフロー図の中から洗い出します。

例として「図 3.2.3 データフロー図」自動車生産ラインシステムで使用するファイルを洗い出してみます。

［生産管理 ASTMAC］
- 本日生産計画ファイル　（生産指示管理タスクで書き込み）
- 組立実績ファイル　（組立指示管理タスクで書き込み）

［検査管理 ASTMAC］
- 検査情報ファイル　（検査指示管理タスクで書き込み）

3.4.2 ファイル形式の決定

ASTMAC システムで以下のファイル形式を選択する必要があります。

1. データベース
   大量なデータを保存します。専用のデータベースソフトが必要です。
   PC もデータベース専用の PC を用意してください。ASTMAC システムと同一の PC 上で稼働することは推奨しません。

2. ISAM ファイル形式
   ISAM ファイルは ASTMAC に提供されているレコードの高速検索に適したファイルで，内部的には表形式のテキストデータを格納できる構造になっています。格納されているデータは ASTMAC の AP フォームの VBA からアクセスが可能です。
   ASTMAC では ISAM 用のデータプロバイダのみを標準でサポートします。
   ISAM 用データプロバイダと，Microsoft 社の ADO 2.5 の組み合わせにより，ISAM が提供する数々のレコードアクセス機能を提供します。

3. 通常ファイル形式
   VB および VBA で通常利用するファイル形式です。CSV 形式，INI 形式，テキスト形式等の種々のデータ形式があります。

ASTMAC システムにおいてファイル形式を選択する場合には，以下のように選択することを推奨します。

ファイル形式		目的・使用方法	備考
データベース		大量のデータを保存し，検索を行って，データを自由に参照できます。	
ISAM ファイル形式		検索を行って，データを自由に参照することができるファイルです。ただし，検索のためのキーは決めておく必要があります。	
通常ファイル形式	INI 形式ファイル	固定の設定情報などプログラム起動時に 1 回だけ読むような場合に使用します。書き込み，編集は，Windows 標準機能の ”メモ帳” で行います。	
	CSV 形式ファイル	‘，’で区切られたテキストファイルです。“Excel”，“メモ帳”で参照が可能です。	

他にも種々のファイル形式はありますが，必要に応じてファイル形式を選択してください。

3.4.3 ファイル構造の決定

ファイル構造を決めるには，データタイプ，レコード数，レコード長等を検討します。本章では ASTMAC で提供している ISAM ファイルの設計についてのみ述べます。

ISAM ファイルの仕様は下記のとおりです。

全般仕様

項目	内容
最大オープンファイル数	255／システム
最大ファイルハンドル数	32／プロセス
最大レコード長	4096 バイト
最大レコード数	500,000 レコード
最大フィールド数	40 フィールド
フィールドタイプ	符号付き 16 ビット整数（配列指定可）
	符号付き 32 ビット整数（配列指定可）
	単精度実数（配列指定可）
	倍精度実数（配列指定可）
	テキスト（文字列）
レコード型	固定長
ファイルタイプ	標準／ハイブリッド／メモリ

インデックス仕様

項目	内容
最大キー数	8
最大キー長	128 バイト
インデックスタイプ	単一インデックス／複合インデックス
インデックス属性	ユニーク／重複

ISAM ファイルの設計は基本的に以下の手順で行います。

1. 使用するフィールドの洗い出し（最大 40 フィールド）
2. 各フィールドのデータタイプ（整数／実数／文字列）
3. 各フィールドのフィールド名を決定する
4. インデックスキーを決める
5. 使用する最大レコード数を決定する（最大 500,000 レコード）
6. ISAM ファイルのファイルタイプ（標準／ハイブリッド／メモリ）を決める

インデックスキーの決め方は，一般的に次のように行います。
・検索に使用するキーはすべてインデックスキーにします。
・常に 2 つ以上のキーを一緒にして検索するような場合に，複合インデックスを設定します。

ファイルタイプの決め方は，一般的に次のように行います。
・通常は，標準に設定します。
・標準タイプよりも，検索スピードをアップさせたいときは，ハイブリッドにすることも有効です。ただし，インデックスの部分をすべて共有メモリ上に確保するので，十分なメモリが必要です。
・更に高速なレコードアクセスを行う場合，ファイルタイプをメモリにすることもできます。ただし，すべて共有メモリ上に作成されるので，電源 OFF（またはファイルクローズ）でファイルは削除されます。また，更に大量なメモリが必要です。

3.4.4 SAMファイルの設計例

下記に ISAM ファイルの例を示します。このファイルの設計例も合わせて記述します。

・ ISAM ファイルの例

フィールド
↓

部品コード	棚番号	在庫数
DFG0567	B23	56
ASTM121	A11	45
SD02	A01	78
UT6700	A11	89
KJY012	U10	67
IUY998	U56	120
SD01	B56	66
SD01	H23	45
SD01	I56	66
XP0202	Z99	25
UT390	C44	55

←フィールド名または列名

←レコード

・ ISAM ファイルの全般仕様の例

項目	内容
レコード長	34 バイト
最大レコード数	1,000 レコード
フィールド数	3 フィールド
ファイルタイプ	標準
キー数	2

・ ISAM ファイルのフィールド定義の例

フィールド名	フィールドタイプ	フィールド長	インデックスキータイプ／属性	備考
部品コード	テキスト	20 バイト	単一／ユニーク	
棚番号	テキスト	10 バイト	単一／重複	
在庫数	32 ビット整数	4 バイト	-	

3.4.5 ISAMファイルを使用する場合の注意点

ISAM ファイルを使用する場合の注意点を示します。

① 書き込み（追加保存，置き換え保存）を行う場合は，1 つのタスクから行うことを推奨します。
② メモリ型／ハイブリッド型は，ファイルをオープンしたときにファイルまたはインデックスブロックがメモリ上に作成されます。このため常に稼動しているタスクが起動時にオープンしてください。また，インデックスをメモリに展開するのに時間がかかります。
③ ファイルサイズは，実際に標準タイプで作成して，できたファイルサイズを確認してください。
④ メモリサイズは，メモリ型の場合，一旦標準タイプで作成したファイルのサイズがメモリ上に必要だと考えてください。ハイブリッド型の場合も，一旦標準タイプで作成したファイルのサイズからハイブリッド型で作成したファイルサイズの差分がメモリ上に必要になります。

3.5 I/Oオブジェクト設計

コントロールオブジェクトと密接に関係するのが I/O オブジェクトです。本章では I/O オブジェクトの詳細について記述します。

I/O オブジェクトとは，PLC（プログラマブルロジックコントローラ）などの I/O 機器のデバイスデータを取り込み，データサーバ上のコントロールオブジェクトにデータを設定したり，コントロールオブジェクトに設定されたデータを I/O 機器のデバイスに書き込んだりするためのオブジェクトです。また I/O オブジェクトは，I/O 機器からの非同期イベントを受信し，ユーザプログラムに非同期イベントの受信を通知することもできます。

I/O オブジェクトは I/O プラットフォームと I/O ドライバにて構成されます。
I/O プラットフォームは，データサーバ内のコントロールオブジェクトとデータ交換を行い，I/O ドライバの働きを制御します。I/O ドライバは，I/O 機器と通信し，実際に I/O デバイスをアクセスする機能を担います。
I/O オブジェクトは図 3.5.1 で示す構成となっています。
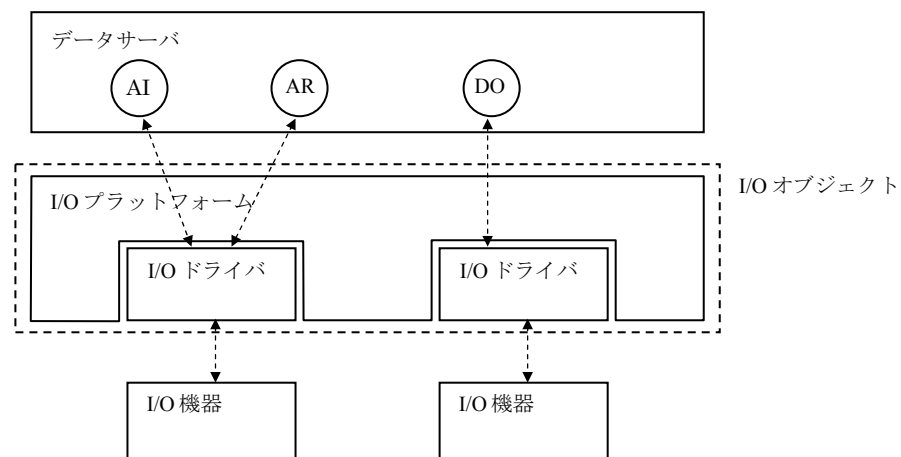
図 3.5.1 データサーバと I/O オブジェクトの関係

この I/O オブジェクトを用いることにより，システムを構築するユーザは I/O 機器との複雑な通信手順をアプリケーションプログラムで作成する必要が無く，データサーバ上のコントロールオブジェクトにて簡単に I/O 機器のデバイスデータをアクセスすることが可能になります。

I/O プラットフォームには，I/O ドライバを差し込む口としてのソケットが，ベースタイプでは 32 個用意されています。このソケットに I/O 機器と通信するための I/O ドライバを差し込んで使用します。ただし，I/O ドライバとコントローラは必ず 1 対 1 で対応しなければならないため，同種の I/O 機器（例えば FA-M3）を 2 台接続する場合には，I/O プラットフォームが用意する 2 つのソケットに，同じ I/O ドライバを別々に差し込む必要があります。

I/O ドライバにどのようなものがあるのかを「表 3.5.1 I/O ドライバの種類」に示します。

表　3.5.1 I/O ドライバの種類

I/O ドライバ	説明
FA-M3 Ethernet 接続 FA-M3 RS-232-C 接続	コントローラに FA-M3 を使用する場合に必要な I/O ドライバです。 ASTMAC の標準 I/O ドライバとして提供されています。
MELSEC Ethernet 接続 MELSEC RS-232-C 接続	コントローラに MELSEC を使用する場合に必要な I/O ドライバです。 MELSEC 接続パッケージのライセンスを登録することで提供されます。
YEWMAC	YEWMAC と接続する場合に必要な I/O ドライバです。 YEWMAC 接続パッケージのライセンスを登録することで提供されます。 1E フレームプロトコルでの通信をサポートしています。
SYSMAC	コントローラに SYSMAC を使用する場合に必要な I/O ドライバです。 SYSMAC 接続パッケージのライセンスを登録することで提供されます。 FinsGateway がサポートする様々なオムロン FA ネットワークに対応します。
STARDOM Controller(*1)	FCJ/FCN と接続する場合に必要な I/O ドライバです。ASTMAC の標準 I/O ドライバとして提供されています。
OPC	OPC サーバ と接続するための I/O ドライバです。
カスタムドライバ(*2)	ユーザが VB で独自に開発し，I/O ドライバとして提供することができます。 カスタムドライバ接続パッケージのライセンスを登録することで使用することができるようになります。秤量器や温調計などの計器や，他社 PLC などを接続する場合に有効です。
MELSEC EZSocket	MELSEC と EZSocket を経由して接続する場合に使用します。
DARWIN	DARWIN と Ethernet により接続する場合に使用します。
DAQSTATION	DAQSTATION と Ethernet により接続する場合に使用します。
電力モニタ	電力モニタと RS- 232- C/RS- 485 コンバータを経由して接続する場合に使用します。

*1：　R4 よりサポートされています。
*2：　動作環境仕様，サポート機能など，一部変更あり

I/O プラットフォームに差し込むための I/O ドライバは，オブジェクトビルダを起動し，I/O オブジェクトのフォルダの中に登録します。接続する I/O 機器のホスト名（IP アドレス）や RS-232-C 接続であれば通信ポート等のを I/O ドライバのプロパティとして設定することで，デバイスデータのアクセスが可能になります。

●I/O オブジェクトを設計するにあたってのガイド

I/O オブジェクトを設計するにあたって，システムを構築しようとする ASTMAC にどのような I/O 機器が繋がれているかをシステム構成図にて確認します。以下に LAN に接続されている PLC を例にとって説明します。
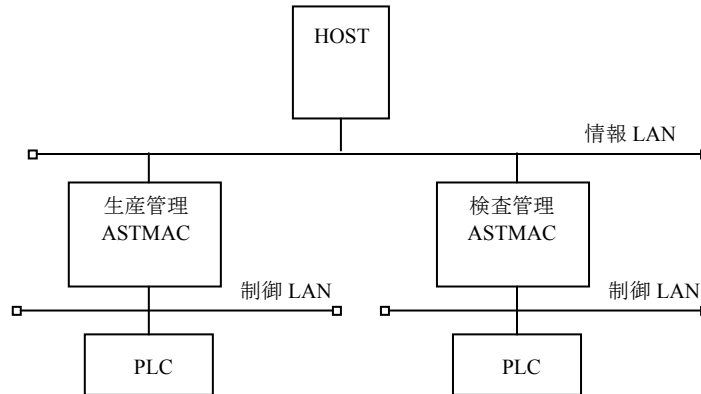
図　3.5.2 ASTMAC ネットワーク構成例

制御 LAN に繋がれている PLC は，標準またはオプションで I/O ドライバが提供されているものであれば，I/O ドライバを I/O オブジェクトに登録します。また，ASTMAC で I/O ドライバが用意されていない PLC である場合，カスタムドライバを作成することも可能です（開発キットが必要）。また，作成したカスタムドライバは I/O オブジェクトに登録し，他の I/O ドライバと同等に取り扱うことができます。カスタムドライバを I/O オブジェクトとして登録する場合は，カスタムドライバ接続パッケージのライセンスが必要です。
他には，アプリケーションフォームにて PLC 通信用のプログラムを作成し，デバイスデータをアクセスする方法もあります。

必要な I/O オブジェクトの選択が完了した後，各装置に Ethernet の IP アドレスを割り振ります。IP アドレスはネットワークアドレスとホストアドレスに分けられます。ネットワークアドレスは，同一 LAN 上のすべての装置で同じアドレスにし，各 LAN 毎に異なる値を設定します。ホストアドレスは，同一 LAN 上で必ずユニークとなるよう，アドレスを割り振ります。この時，PLC はホストアドレスを 1 番から，ASTMAC は 100 番から割り振るようにしておくと，後で PLC などが追加となった場合などにアドレスが混乱することを防ぐことができます。
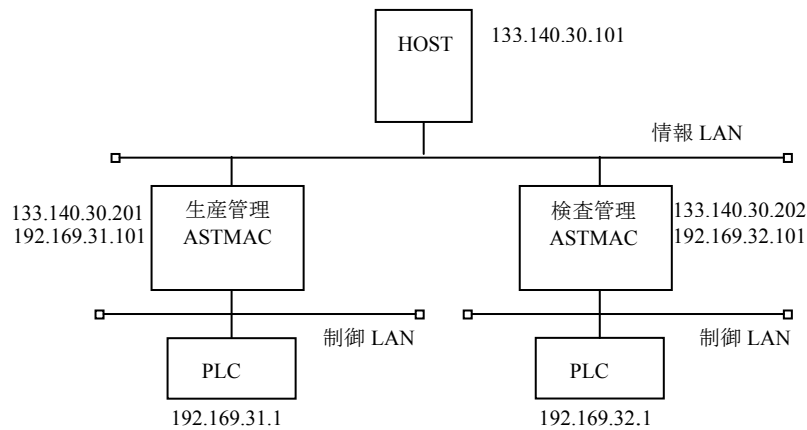
図　3.5.3 ASTMAC ネットワーク構成例

上記のように IP アドレスを決め，各々の装置に対し IP アドレスを設定します。IP アドレスの設定方法については各機器固有の設定手順に従い設定してください。
接続する I/O 機器の種類，IP アドレスが決定すれば，I/O オブジェクトの定義に移ります。

● I/O オブジェクトの定義ガイド

I/O オブジェクトはオブジェクトビルダで定義します。
I/O オブジェクトのプロパティ設定は以下に示す内容を目安に行ってください。

① 「ドライバ共通」タブ－ 監視時間，リトライ回数，装置タイプ
このタブのプロパティ値は，I/O オブジェクトを追加したときに表示されるデフォルト値のまま使用します。特に変更する必要はありません。
ただし，システム操業状態で I/O ドライバの通信異常が発生するようであれば，監視時間，リトライ回数を変更し，通信異常が発生しないところで調整してください。
装置タイプは接続するパッケージの取扱説明書を参照してください。例えば，装置タイプは次の種類があります。
Passive：　ASTMAC からのコマンドに対し，レスポンスを送信する機能持った装置です。
Active：非同期にデータを送信する機能を持った装置です。
Hybrid：Passive と Active の両方の機能を兼ね備えた装置です。

② 「ドライバ固有情報」タブ－[ホスト名]，[ターゲット情報]，[ホスト情報]
上記設定項目は Ethernet 接続を選択した場合に表示されます。
ホスト名には，登録している I/O ドライバが通信を行う I/O 機器を特定するため，該当する I/O 機器の IP アドレスを指定します。
または，Windows のシステムファイルである Hosts
（C:¥Winnt¥system32¥drivers¥etc¥Hosts のこと）にホスト名と IP アドレスを定義していれば，ホスト名を指定することもできます。
ターゲット情報は，コマンドレスポンス方式でデータのやり取りを行う通信ポート番号とプロトコルを設定し，ホスト情報は，非同期で I/O 機器からのデータを受信する通信ポート番号とプロトコルを設定するものです。
I/O ドライバが FA-M3-Ethernet の場合はこれらの情報を変更する必要はありませんが，その他の I/O ドライバを使用する場合は，接続する I/O 装置の内容に合うように，設定を行います。

※ RS-232-C の場合
「ドライバ固有情報」タブ－[通信速度]，[デバイス名]，[データ形式]
RS-232-C で接続する I/O ドライバを選択した場合は，ドライバ固有情報のタブが上記の項目に変わります。ここには既にデフォルトの値が登録されていますので，変更する必要はありません。I/O 機器の設定を表示された設定します。
デバイス名は｢COM1｣がデフォルトとなっていますが，必要に応じて変更します。

③ 「その他」タブ
I/O 機器固有の情報を設定するためのページです。
本ページは I/O 機器の取扱説明書に基づき最適な値を設定します。

3.6　コントロールオブジェクトの選択

ASTMAC システムがデータを扱う場合，コントロールオブジェクトを利用します。コントロールオブジェクトは，オブジェクト指向データサーバ（以下，データサーバと呼びます）上に配置される部品です。
このコントロールオブジェクトは，I/O オブジェクトを利用してフィールド機器とのデータ交換を行ったり，アプリケーションフォームとのデータ交換を行います。
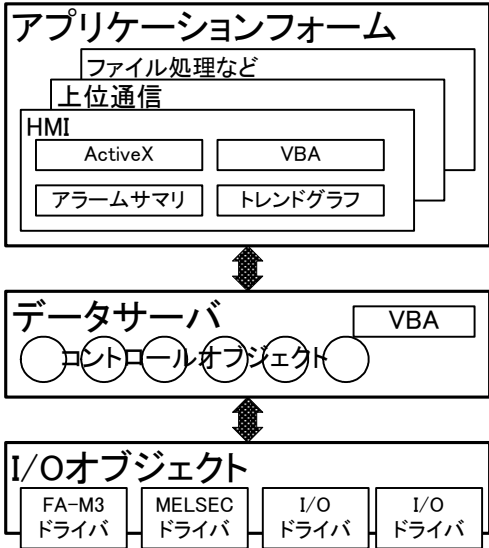
図　3.6.1 ASTMAC の構成

コントロールオブジェクトには，大別して

デバイスタグ オブジェクト	I/O データ入出力を行い，工業量変換やアラーム監視，入力積算等を行います。
アプリケーション支援 オブジェクト	VBA プログラミングを支援するシステムリソースを提供します。

の 2 種があり，それぞれ種々の機能を持った独立した部品が提供されています。
コントロールオブジェクトの設計を行う場合には下記の手順で行います。

1. 使用するコントロールオブジェクトを選択します
2. 各コントロールオブジェクトのプロパティの設定を行います。
3. VBA によりカスタマイズが必要な処理を追加します。（ただし，パフォーマンスに十分注意した設計が必要です。）

これらのコントロールオブジェクト設計が完了したら，オブジェクトビルダを用いて設定を行います。

ASTMAC のコントロールオブジェクトの設定は，オブジェクトビルダで行います。
オブジェクトビルダについての詳細は，「ASTMAC データサーバ解説書（IM 34P02D11-01）」を参照してください。

3.6.1 コントロールオブジェクトの選択

表 3.6.1 コントロールオブジェクト一覧

	タイプ	説明	機能	備考
デバイスタグオブジェクト	DI	ディジタル入力	1 ビットのデータ入力	
	DO	ディジタル出力	1 ビットのデータ出力	
	DR	ディジタルレジスタ	1 ビットのデータ配列入出力	
	AI	アナログ入力	1 ワードのデータ入力	
	AO	アナログ出力	1 ワードのデータ出力	
	AR	アナログレジスタ	1 ワードのデータ配列入出力	
	TXT	テキスト	128 バイトまでの文字データ入出力	
	XAI	拡張アナログ入力	2 ワードのデータ入力	
	XAO	拡張アナログ出力	2 ワードのデータ出力	
	XAR	拡張アナログレジスタ	2 ワードのデータ配列入出力	
	XTX	拡張テキスト	4096 バイトまでの文字データ入出力	
アプリケーション支援オブジェクト	TMR	タイマ	指定時刻，定周期での VBA プログラム起動	
	EVT	イベント	外部アプリケーションからの VBA プログラム起動	
	BUF	バッファ	任意データを格納するバッファ	
	SIG	シグナル交信	プロセス間で事象の発生（シグナル）の送信および受信	マルチタスク支援パッケージ
	BD	ブロックデータ	32K バイトまでのデータ入出力	
	BDA	ブロックデータアクセス	ブロックデータオブジェクトの持つ共有データを高速にアクセス	

この上記コントロールオブジェクトの中から，構築する ASTMAC システムで使用するものを選択します。このコントロールオブジェクトから選択するには，

1. データを授受する相手
- 対 PLC
- ASTMAC のタスク間（AP フォーム間）

2. データサイズ
- 1 ビット
- 1 ワード
- 2 ワード
- ブロックデータ（多量のデータを一括して扱う）

3. データタイプ
- 入力データ
- 出力データ

などを考慮します。配列やブロックデータを使用するなど，できるだけオブジェクトを少なくする方向で設計してください。

基本的なコントロールオブジェクトと，その選択基準を下記に示します。使用する場合には取扱説明書を参照してください。

1. 対 PLC からデータ収集を行う時

機能	タイプ	説明	使用例
1 ビットのデータ入力	DI	ディジタル入力	スイッチの On/Off 状態
1 ビットのデーター括入力	DR	ディジタル入力	アラーム状態監視
1 ワードのデータ入力	AI	アナログ入力	電力使用量
2 ワードのデータ入力	XAI	拡張アナログ入力	搬送車位置情報
1 ワードの配列データ入力	AR	アナログレジスタ	複数機器の電力使用量
2 ワードの配列データ入力	XAR	拡張アナログレジスタ	複数機器の搬送者位置情報
非同期データ	TXT	テキスト	生産開始通知
複数のデータを入力	BD	ブロックデータ	生産実績データ
128 バイトまでの文字データ入入力	TXT	テキスト	AGV への搬送指示
4096 バイトまでの文字データ入入力	TXT	拡張テキスト	AGV への搬送指示

収集周期は，できるだけまとめるようにし，少ない種類にとどめるようにします。

2. 対 PLC へのデータ出力を行う時

機能	タイプ	説明	使用例
1 ビットのデータ出力	DO	ディジタル出力	バルブの開閉制御
1 ビットのデーター括入力	DR	ディジタル入力	複数バルブの開閉制御
1 ワードのデータ出力	AO	アナログ出力	電力使用量
2 ワードのデータ出力	XAO	拡張アナログ出力	搬送車位置情報
1 ワードの配列データ出力	AR	アナログレジスタ	複数機器の電力使用量
2 ワードの配列データ出力	XAR	拡張アナログレジスタ	複数機器の搬送者位置情報
複数のデータを出力	BD	ブロックデータ	生産指示データ
128 バイトまでの文字データ入入力	TXT	テキスト	AGV からの搬送指示
4096 バイトまでの文字データ入入力	TXT	拡張テキスト	AGV からの搬送指示

3. ASTMAC のタスク間（AP フォーム間）でのデータ授受を行う時

機能	タイプ	説明	使用例
非同期データ	SIG	シグナル交信	処理完了通知
複数のデータ入出力	BD	ブロックデータ	生産指示データ
BD のアクセス	BDA	ブロックデータアクセス	

3.6.2 コントロールオブジェクトのプロパティ

コントロールオブジェクトのプロパティは，以下のように分類できます。
本書では主なプロパティのみを解説いたします。
各プロパティの詳細および記載のないプロパティは，取扱説明書を参照してください。
「ASTMAC データサーバ解説書」（IM 34P02D11-01）
「ASTMAC データサーバリファレンスマニュアル」（IM 34P02D11-02）

プロパティのグループ	説明	プロパティ
全般	すべてのオブジェクトに共通して必要な，名前，タイプ，コメント情報を保持するプロパティです。	オブジェクト名：ObjName オブジェクトタイプ：ObjType コメント：ObjCmnt
詳細	I/O のタイプやオブジェクトとの関係を保持するプロパティです。	I/O オブジェクト名：IoName I/O アドレス：IoAd スキャン状態：ScanEnab データ取得方式：ScanProc
ユーザ定義	ユーザが任意のデータを保持する用途で使用できるオブジェクト内のバッファです。	Extra1～Extra3（数値） ExtraStr（文字列）
シミュレーション／トレース	シミュレーションやトレース関連のプロパティです。	シミュレーション状態：SimEnab シミュレーションパターン： トレース状態：TraceEnab

● 全般プロパティ

コントロールオブジェクトの全般プロパティには主に以下のものがあります。

名前	説明	備考
グループ名	英数かな文字，アンダースコア（"_"）で構成される，プロジェクト内でユニークな 15 文字までの文字列	・グループ名.オブジェクト名でプロジェクト内ユニーク ・かなには漢字も含む ・数字で始まるものは除く
オブジェクト名	英数かな文字，アンダースコア（"_"）で構成される，グループ内でユニークな 16 文字までの文字列	

グループはオブジェクトの機能単位で分類することをお勧めします。オブジェクトをグループ化することにより，再利用性の高いオブジェクト定義を行うことができます。
例えば，下図のように WorkSpace の直下にオブジェクトを定義してしまうと，定義情報のエクスポートができません。他のマシンでも同様の定義を行なうのであれば，例えば，AIObject グループに定義し，AIObject の定義情報をエクスポートし，他のマシンでその定義情報をインポートします。

図 3.6.2 再利用のできないオブジェクト定義

図　3.6.3 再利用の可能なオブジェクト定義

エクスポートは［ファイル］の［定義情報エクスポート］で行います。

図　3.6.4 エクスポート

インポートは，［ファイル］の［定義情報インポート］で行います。
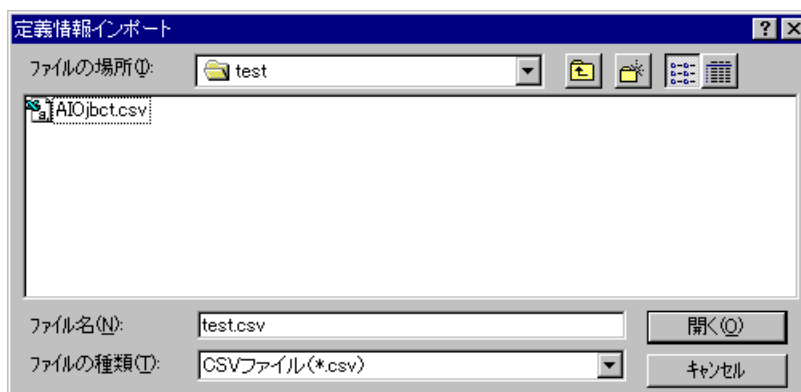
図　3.6.5 定義情報インポート

● 詳細プロパティ

コントロールオブジェクトの詳細プロパティには主に以下のものがあります。

・ I/O オブジェクト名称
・ I/O アドレス
・ 入力モード

［I/O オブジェクト名称］

本書「3.6 I/O オブジェクト設計」を参照してください。

［I/O アドレス］

使用するデバイスにより I/O アドレスを定義する必要があります。下記のドキュメントをご参照ください。

接続機器	ドキュメント	章	備考
FA-M3	ASTMAC データサーバ解説書（IM 34P02D11-01）	3.3.2 詳細タブ	
STARDOM Controller	VDS エンジニアリングマニュアル（IM 34P02D02-01）	A1.2 システム構成手順 FCN/FCJ と接続する	
YEWMAC500	YEWMAC500 接続パッケージ説明書（IM 34P02H03-01）	5.2 I/O アドレス設定	オプションパッケージ
MELSEC	MELSEC 接続パッケージ説明書（IM 34P02G01-01）	PART-B 3.3 I/O ドライバの組み込みと設定（Ethernet） PART-B 4.3 I/O ドライバの組み込みと設定（RS）	〃
SYSMAC	SYSMAC 接続パッケージ説明書（IM 34P02G06-01）	5.2 I/O ドライバの組み込みと設定	〃

［入力モード］

入力モードは，データ収集を行うデバイスタグオブジェクト（DI,AI,XAI 等）のプロパティです。詳細機能について「ASTMAC データサーバ解説書（IM 34P02D11-01） 2.4.3.3 入力モード（ScanProc プロパティ）」を参照してください。

入力モードを選択するときには以下の指針で行います。

入力モード	データ収集	使用方法
変化	定周期	定周期で PLC のデータをモニタする
定周期	定周期	定周期で PLC のデータをモニタし，さらに積算処理等を行う
変化（パートタイム）	定周期(*)	コントローラからのデータ取得が有効の間,更新周期に従ってデータを収集し，収集したデータが前回値と異なる場合のみ，デバイスタグオブジェクトに更新通知を行う
定周期（パートタイム）	定周期(*)	コントローラからのデータ取得が有効の間, 更新周期に従ってデータを収集し，データを収集したタイミングでデバイスタグオブジェクトに更新通知を行う
非同期	非同期	PLC からの非同期通知によりリアルタイムに処理を行う（PLC により使用できるオブジェクトタイプに制限があります）
ワンショット	Refresh メソッド実行時	AP が必要なときだけ PLC からデータ収集を行う（不必要なデータ収集はシステムの負荷を重くします） ただし，定周期の読み込みとは違い I/O アクセスの最適化は行われません

*：　アプリケーションの状態によってコントローラからのデータの取得操作が自動的に制御（ON/OFF）されます。

● アラームプロパティ

あらかじめ定義されたアラーム条件が成立した場合に，オブジェクトがアラーム状態となります。アラームの条件としては，通信異常から上下限のチェックなど数種類ありますので，システムに応じて設定を行ってください。

次に，一般的な使用方法を述べます。

- 通信異常を設定することにより，通信系の異常を検知することが可能になります。
- アナログデータの場合，入力オープンチェックを設定することにより，断線など入力の異常を検知することが可能になります。
- 入力の ON／OFF が切り替わる度に，メッセージを通知したいような場合は，状態変化アラームを使用します。あるいは，ON，OFF いずれかが正常で，逆になったときを異常と判断し，メッセージ通知したいような場合は，正常からの変化アラームを使用します。状態変化アラームと正常からの変化アラームを 1 つのオブジェクトで同時に使用することはできません。
- 上下限アラーム，変化率アラーム，積算上限アラームは，システムの仕様に基づいて，使用してください。

詳細は，「ASTMAC データサーバリファレンスマニュアル（IM 34P02D11-02） 2.6 プロセスアラーム判定」を参照してください。

● 積算プロパティ

積算機能には，次の 3 種類があります。いずれも取得方式は，定周期のみとなります。

- アナログ入力積算
  AI オブジェクトの場合，現在値を時間単位で積算します。
  電力量の積算値などを取得する場合に有効です。
- 稼動時間積算
  DI オブジェクトの場合，ON／OFF どちらかの稼動時間を積算します。
  機械の稼動時間を取得するような場合に有効です。
- 変化回数積算
  DI オブジェクトの場合，ON／OFF どちらかの回数を積算します。
  完成品の個数などを求める場合に有効です。

詳細は「ASTMAC データサーバリファレンスマニュアル（IM 34P02D11-02） 2.5 入力積算」を参照してください。

● シグナルコンディション

デバイスタグオブジェクトは，コントローラから入力した生のデータを工業量に変換して現在値（Cv）とします。データ出力時は，工業量からコントローラのデータ型への変換を行います。この変換操作をシグナルコンディションといいます。
標準では，上下限を指定してリニア変換する機能が 7 つの数値タイプ分用意されています。その他に，数値の範囲と上下限を任意に指定できるユーザ定義タイプがあります。ユーザ定義タイプは 256 個まで定義できます。
FA-M3 の入力レンジが-20000～20000 の AI モジュールを使用して，0～1000 に変換するような場合は，入出力範囲を-20000～20000，上下限を 0～1000 に指定したユーザ定義タイプを使用します。
シグナルコンディションタイプには，リニア変換タイプと VBA マクロを使用する汎用レンジ変換タイプがあります。リニア変換タイプは，アナログ入出力を行うデバイスタグオブジェクト（AI，AO，AR，XAI，XAO，XAR）で使用することができます。
汎用レンジ変換タイプは，アナログ入力オブジェクト（AI）でのみ使用することができます。
コントローラ側で，変換を行う場合は特に使用する必要はありません。

詳細は，「ASTMAC データサーバリファレンスマニュアル（IM 34P02D11-02） 2.3 シグナルコンディション」を参照してください。

● タイマオブジェクト

タイマオブジェクトには，は以下の 3 つの機能があります。各々について記述します。

- 継続タイマ
  設定された周期ごとに，継続してイベントを発行します。
  定周期で，演算をするような処理を作成する場合に使用します。
- ワンショットタイマ
  設定された周期が経過したとき，1 回だけイベントを発行したあと，停止します。
  オペレータがボタンクリック後，一定時間たったら運転を開始（接点 ON）するなどのような処理を行うときに使用します。
- 指定時刻起動
  指定された時刻になったとき，1 回だけイベントを発行して停止します。
  1 日ごとにデータ保存するような処理の場合に，時刻を指定して使用します。

詳細は「ASTMAC データサーバリファレンスマニュアル（IM 34P02D11-02） 3.2.1 タイマオブジェクト」を参照してください。

● ブロックデータ

ブロックデータを使用する場合は，マルチタスク支援パッケージが必要になります。ブロックデータオブジェクトにより，タスク間のデータ共有を簡単に行なうことが出来ます。
ブロックデータオブジェクトは，データサーバ内に最大 32Kbyte までの共有データを保持することができます。ブロックデータの特徴は次の通りです。

- データサーバ上に複数貼り付けることができます。
- アプリケーションフォームからブロックデータアクセスオブジェクトを使用してアクセスすることができます。
- ブロックデータオブジェクトに保持されるデータは構造化されていて，すべて名前でアクセスできます。
- ネットワークでつながれた他のコンピュータ上で動作しているデータサーバ上のプロックデータオブジェクトをアクセスすることが可能です。

また，ブロックデータオブジェクトは I/O 機器との大量データ送受信することが可能です。ブロックデータオブジェクトを使用することにより，I/O 機器に対して，最大 32KByte までのデータを一括して，または一部分だけ，アプリケーションの任意のタイミングで，送受信の通信手順を意識せずに行うことができます。この機能により，ブロックデータオブジェクト内の共有データが，I/O 機器に送信されます。逆に，I/O 機器から読み込んだデータでオブジェクト内の共有データを書き換えることができます。I/O 機器から読み込んだデータをブロックオブジェクト内の共有データに書き込みますので，ブロックデータアクセスオブジェクトを用いて他のプロセスからアクセスすることが可能です。

詳細は「ASTMAC マルチタスク支援パッケージ説明書（IM 34P02H04-01） 2.2 データ共有，2.3 I/O 機器との大量データの送受信」を参照してください。

● シグナル

シグナル機能を使用するためには，マルチタスク支援パッケージが必要になります。シグナル機能を使用する場合は，標準の AP 支援オブジェクトのイベントオブジェクトと混在しないようにします。標準のイベントオブジェクトは，データサーバ上の VBA にだけイベントを発信できますが，それ以外はできないので，タスク間のタイミングを自由にとるためにはマルチタスク支援パッケージを導入してください。

ブロックデータとシグナルにつては，本書の「3.7.5.プログラム間のデータ共有」で詳しく説明します。

3.6.3 データサーバ上のVBA

データサーバには VBA が搭載されています。データサーバが収集したデータをもとに標準機能に無い処理をユーザが簡単に構築することができます。VBA でプログラミングするときの主な目的には以下のようなものがあります。これまでの説明を参考に，プログラミングしてください。

・ユーザ独自のシグナルコンディション
・アラーム処理
・非同期処理
・タイマ処理
・ブロックデータ読み込み／書き込み処理

コントロールオブジェクトのイベントには，以下のものがあります。詳細については「ASTMAC データサーバリファレンスマニュアル （IM 34P02G11-01） 6.コントロールオブジェクトのイベント」を参照してください。

デバイスタグオブジェクト

イベント	イベントプロシージャ名	コントロールオブジェクト										
		DI	DO	DR	AI	AO	AR	TXT	AI	AO	AR	XTX
データ変化	CvChange	○		○	○		○	○	○		○	○
アラーム変化	AlmChange	○			○				○			
読み込み完了	ReadComplete											
書き込み完了	WriteComplete		○	○		○	○	○		○	○	○
データリフレッシュ完了	RefreshComplete	○		○	○		○	○	○		○	○
タイマ	Timer											
ユーザイベント	Event											
シグナル	SignalRecived											

アプリケーション支援オブジェクト

イベント	イベントプロシージャ名	コントロールオブジェクト					
		TMR	EVT	BUF	SIG	BD	BDA
データ変化	CvChange						
アラーム変化	AlmChange						
読み込み完了	ReadComplete					○	○
書き込み完了	WriteComplete					○	○
データリフレッシュ完了	RefreshComplete						
タイマ	Timer	○					
ユーザイベント	Event		○				
シグナル	SignalRecived				○		

データサーバ VBA では，外部のコンポーネント（VBE ［ツール］ー［参照設定］で参照可能なオブジェクト）も使用することができます。ただしデータサーバエンジンの動作を遅延させないための十分な注意が必要です。
データサーバの遅延を避けるためにもなるべく使用を避けてください。
また，長期的に VBA を使用してアクセスを行なっているとデータサーバが強制的に切断します。
（詳細は，「ASTMAC データサーバ解説書（IM 34P02D11-01） 2.8 汎用コンポーネントの利用」を参照してください。）

3.7 画面設計

データフロー図よりタスク，ファイル，コントロールオブジェクトの機能／仕様は決定しました。次に画面の設計を行います。
ASTMAC での画面作成は，グラフィックビルダで行います。このグラフィックビルダでは，画面作成をビジュアルに簡単に行うことができます。
ASTMAC の画面作成は，以下の手順で行います。
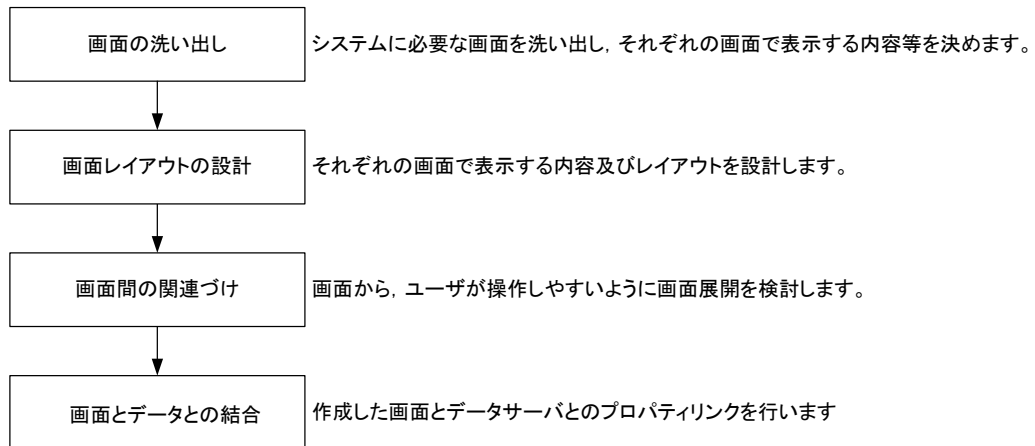
図　3.7.1 画面設計フロー

3.7.1 画面の洗い出し

画面として必要なものを洗い出します。
一般に画面としては，以下に示すような種類があります。

- メニュー画面
- 運転操作／監視画面
- データメンテナンス画面
- トレンド表示画面
- アラームメッセージ履歴画面
- 生産実績一覧画面

など

3.7.2　画面レイアウトの設計

次に，画面のレイアウトを設計します。
グラフィックビルダには，

- 基本オブジェクト
- 描画機能
- 高機能グラフィカル ActiveX コントロール
- シンボルライブラリ
- イメージ
- アニメーション機能
- トレンド表示
- アラーム／メッセージ表示

の機能があり，簡単に高機能な画面を設計&作成できます。
画面の表示内容は，千差万別です。そこで，いくつかの表示サンプルを紹介します。
これらを組み合わせて，見やすくてわかりやすい画面を設計してください。

■ 基本オブジェクト

基本オブジェクトの表示例を以下に示します。
基本オブジェクトは VB の持つコンポーネントも使用することができます。
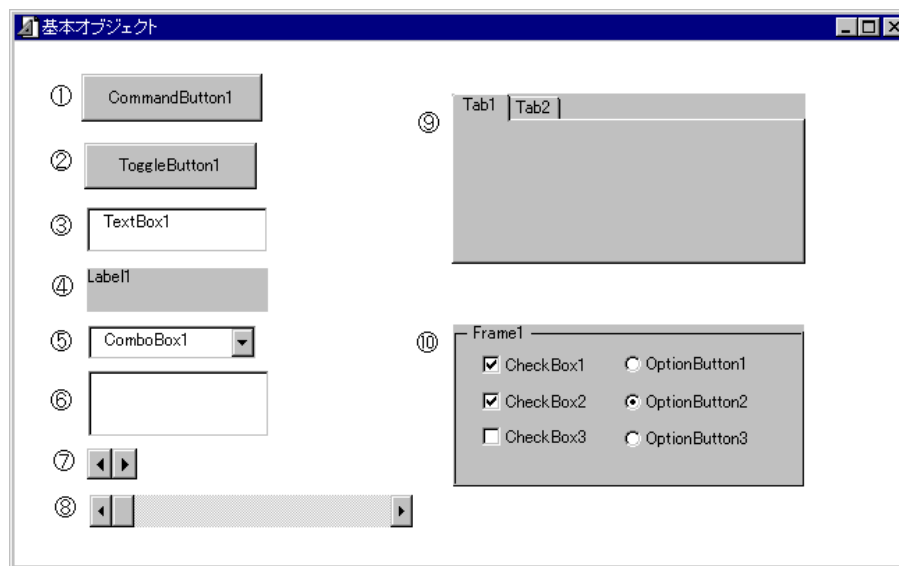
図　　3.7.2 基本オブジェクト

図中番号	コントロール名	目的
①	CommandButton	ボタンをクリックして，何か処理を行う場合に用います。
②	ToggleButton	ボタンをクリックして，選択するような場合に用います。
③	TextBox	設定値などの入力を行う場合に用います。
④	Label	画面上に固定で文字を表示する場合に用います。
⑤	ComboBox	入力するものを複数の選択肢から選ぶ場合に用います。
⑥	ListBox	複数の選択肢から選択する場合のみ，使用します。
⑦	SpinButton	1 つ前のもの，1 つ次のものに移動するような場合に用います。
⑧	ScrollBar	1 度に全部表示できないような場合に，表示部を移動するときに用います。
⑨	Tabstrip	ページをめくるようなイメージにしたいとき，使用します。
⑩	Frame	二者択一，有り／なし　などの選択を行う場合に使用します。

■ 描画機能

ASTMAC は線や図形，文字，ビットマップ等の画像を自由に描画できます。
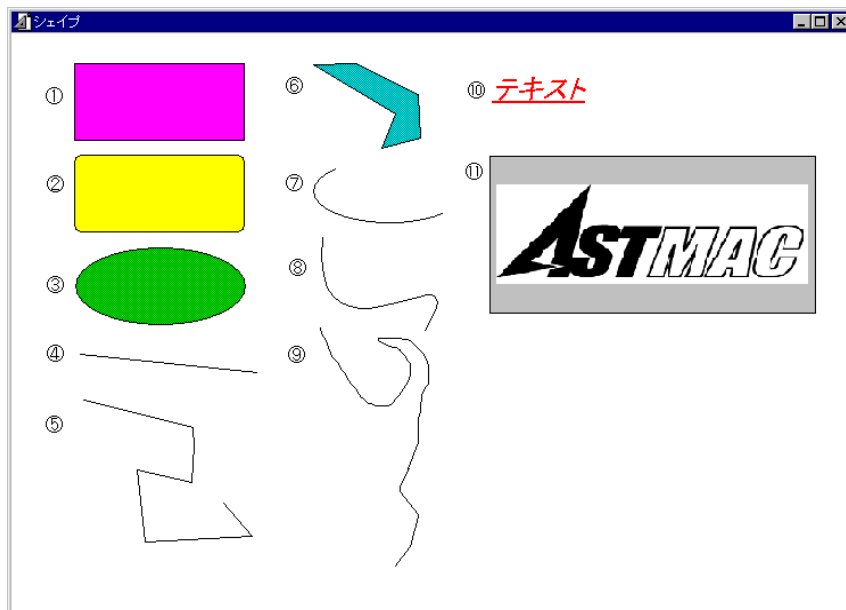
図　3.7.3 シェイプによる描画

シェイプのコントロールとしては，次のようなものがあります。

図中番号	コントロール名
①	四角形
②	角丸四角形
③	楕円
④	直線
⑤	多角直線
⑥	塗りつぶし多角直線
⑦	円弧
⑧	ベジェー曲線
⑨	フリーフォーム
⑩	テキスト
⑪	ビットマップ

図に示すようにシェイプを組み合わせることにより，以下のような描画ができます。これをグループ化してひとつのオブジェクトとすることが可能です。詳しくは，「3.7.2.7 グラフィックオブジェクトの結合」で説明します。
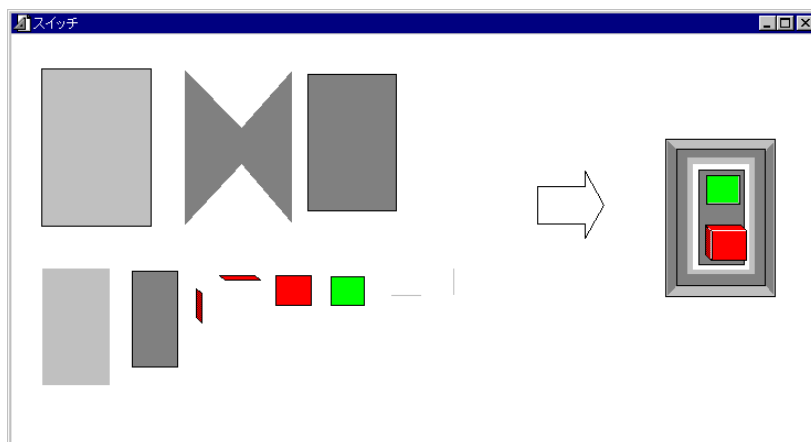
図　3.7.4 スイッチの絵

■ 高機能グラフィカル ActiveX コントロール

ActiveX コントロールを利用すると，「文字」「グラフィック」「画像」が簡単に表示することができます。他の方法でも同等の機能を実現できますが，基本的にはActiveX コントロールを使用してください。
以下にいくつかの代表的な ActiveX コントロールの表示例を示します。
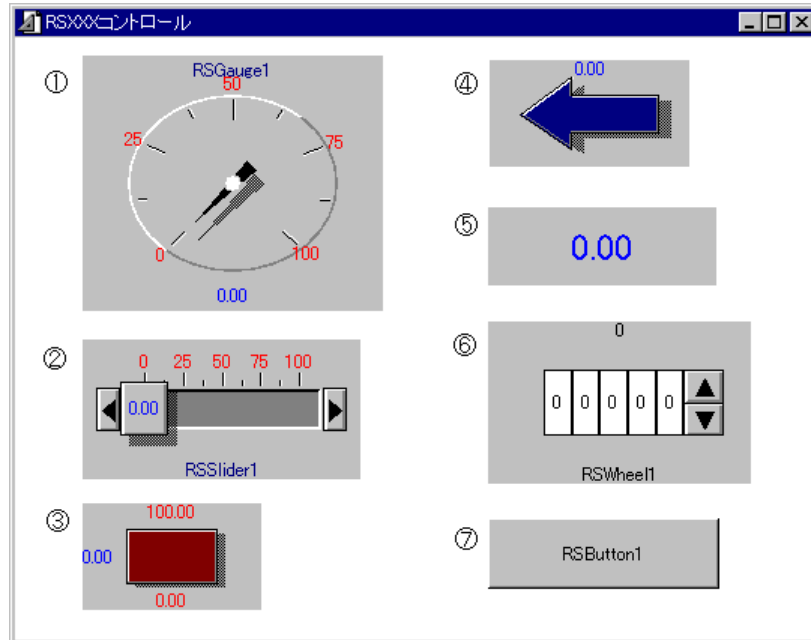
図　3.7.5 ActiveX コントロール

図中番号	コントロール名	説　　明
①	RSGauge	数値データをグラフィカルに表示する場合に使用します。
②	RSSlider	最小値，最大値の設定と豊富なスケール機能により，グラフィカルなデータ入力制御を行うことができます。
③	RSVessel	塗りつぶしが可能で，タンクなどの表示をする場合に使用できます。
④	RSCompare	現在の値と直前の値，チェック値などと比較することが可能です。
⑤	RSData	単純にデータを表示するためのものです。
⑥	RSWheel	カウンタなどの表示，設定などを行う場合に使用します。
⑦	RSButton	標準のボタンよりも高機能で，モメンタリ，トグル，ロッカー，チェックボックス，オプション，デュアルボタン，ダイオード，ピクチャがあります。また，アップ／ダウンおよび値の読み取り，または書き込みといった機能があります

このように種々のコントロールがあります。さらに，プロパティを変更するだけで色々な形で表現できます。「図 3.7.6 RsGauge コントロール一覧（例）」を示します。
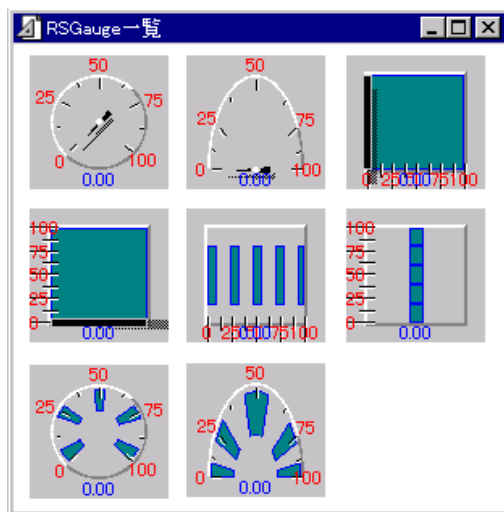
図　3.7.6 RsGauge コントロール一覧（例）

また，RSData，RSButton は，1 つのオブジェクトを配列としても使用できます。例えば，RSButton コントロールをブール型の Excel データベースにリンクして，データベースから読取った値に基づいて複数のボタンがアップまたはダウンのいずれかに設定された配列を作成します。単純な使用方法は，オブジェクトのプロパティ（ファンクションキーF4 で表示）の Value にカンマで区切った複数の値を設定します。詳細は，「ASTMAC グラフィックコントロールリファレンスマニュアル（IM 34P02E01-02）」を参照してください。

■ シンボルライブラリ

ASTMAC には，シンボルライブラリとして多種多様の絵／図が提供されています。これらのシンボルライブラリは，新規に作成することもでき，部品として再利用することが可能です。シンボルライブラリを組み合わせることにより，更に複雑な絵を作ることも可能です。
下記に，ASTMAC が提供しているシンボルライブラリの例を示します。
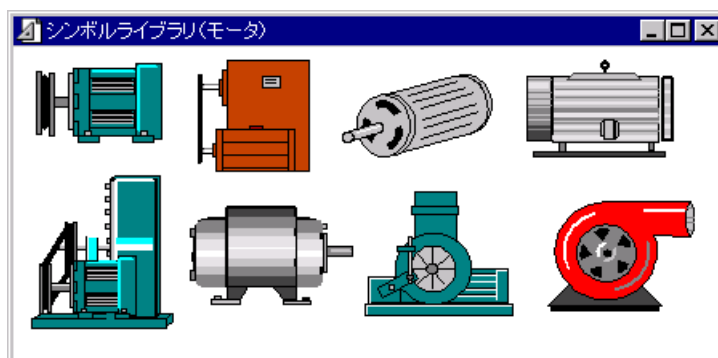
図　3.7.7 シンボルライブラリ（モータ）

その他のシンボルライブラリは，添付資料を参照してください。

■ イメージ

ユーザがディジタルカメラ／スキャナー等から取り込んだ画像をイメージとして表示することもできます。イメージとして取り込めるファイルは以下のものです。

ファイルのタイプ	インポートしたタイプ
.bmp（ビットマップ）	.bmp ファイル
.ico（アイコン）	.bmp ファイル
.wmf（Windows メタファイル）	Graphic オブジェクト

イメージの例を次に示します。

図　3.7.8 イメージの例

■ アニメーション機能

ASTMAC に提供されているシンボルライブラリに登録されている絵やユーザが独自に作成した絵を動画として表示することができます。
アニメーション機能は

- オブジェクトのアンカー接続処理
- オブジェクトの移動処理
- オブジェクトの回転処理
- オブジェクトのリベール機能

をグラフィックに対して行うことができます。

● オブジェクトのアンカー接続

適切な Graphic オブジェクトとアンカー接続を行います。アンカー接続により，複数のオブジェクトを接続させ，あるオブジェクトを別のオブジェクトに接続させながら動かすことができます。アンカー接続した接続点を基点とし（設定によりポイントをずらすことも可能），連動して動かすことができます。
プロパティとしては，Anchor，AnchorX，AnchorＹ等を使用します。
例えば，Anchor を使用する場合は，アンカー接続点に対してオブジェクトが移動する際の角度を設定します。
次にロボットのアームをアンカー接続した例を示します。詳細なテスト方法は「ASTMAC グラフィック機能解説書（IM 34P02E01-01） 6.Grapihc オブジェクトの作成と操作」を参照してください。
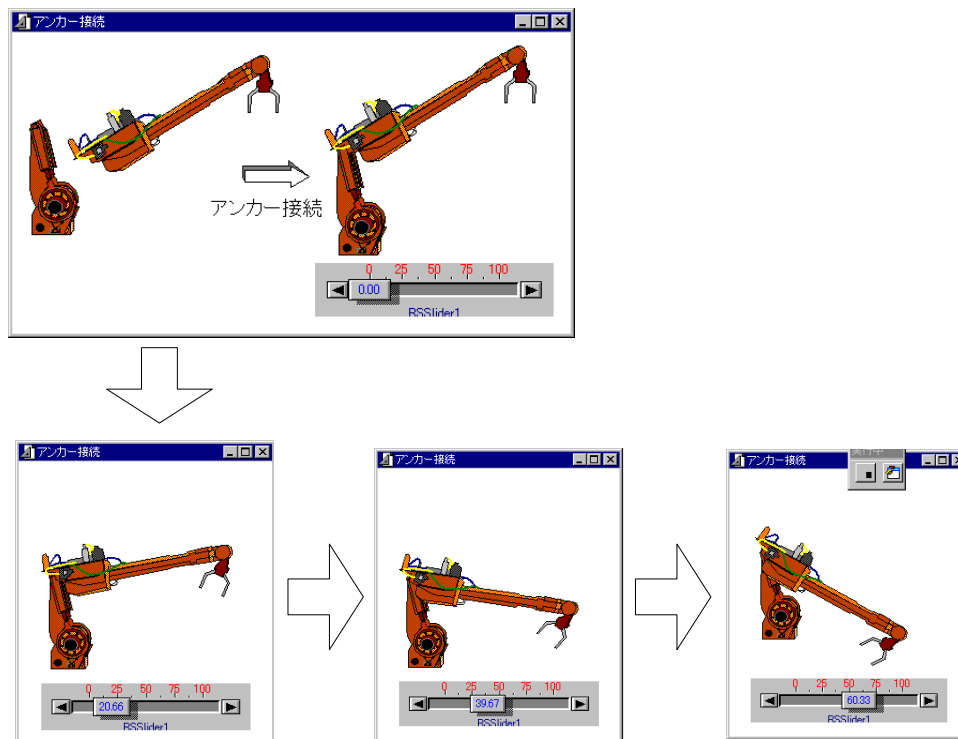
図　3.7.9 アニメーション機能（アンカー接続）

●オブジェクトの移動処理

オブジェクトをデータの変化に応じて，移動させることができます。
プロパティで移動範囲，間隔などを設定することにより，実現できます。そのとき，垂直移動の設定としては，画面の一番上が最小値で，下方向に大きくなっていきますので，設定時に気をつけてください。
プロパティとしては，HSliderStart，HSliderEnd，HSliderSteps，VSliderStart，VSliderEnd，VsliderSteps 等を使用します。
以下にフォークリフトのフォークを上下させるアニメーションの例を示します。
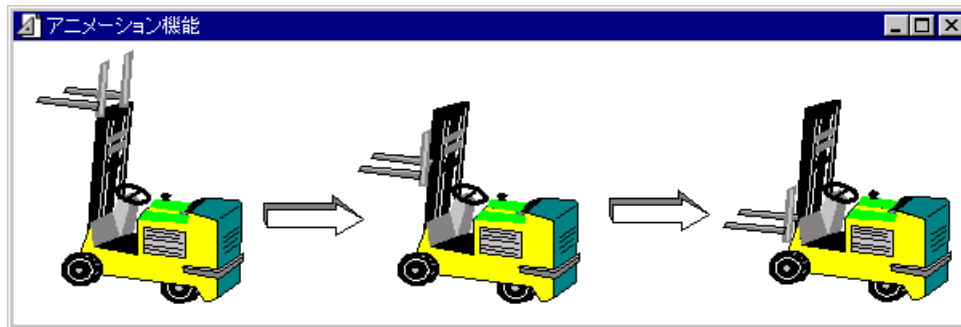
図　3.7.10 アニメーション機能（移動処理）

●オブジェクトの回転処理

オブジェクトをデータの変化に応じて回転させることができます。
プロパティとしては，Rotation，RotationＸ，RotationＹ等を使用します。
以下にタンクローリーを右に 90 度回転させるアニメーションの例を示します。
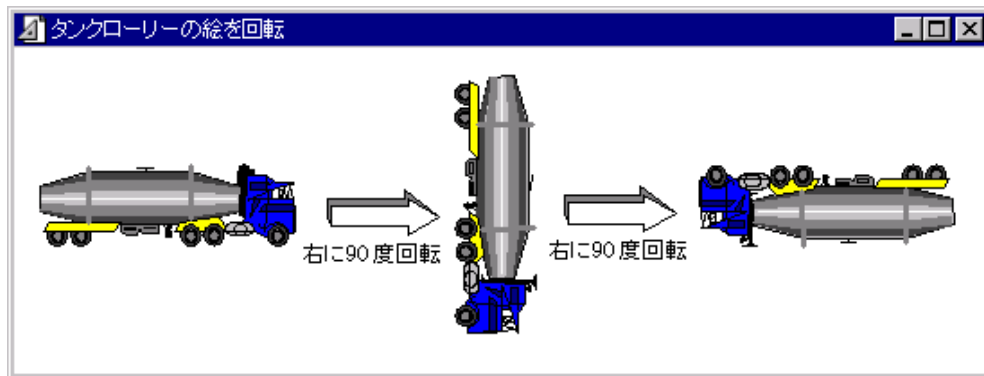
図　3.7.11 アニメーション機能（回転処理）

● オブジェクトのリベール機能

実行時に Graphic オブジェクトを徐々に表示させることができます。これをリベール機能と呼びます。例えば，タンク内部の液体レベルの上昇や降下を表現する場合に有効です。
プロパティとしては RevealMode，RevealPercent を使用します。
以下にタンク内部の液体を上昇させた場合の例を示します。
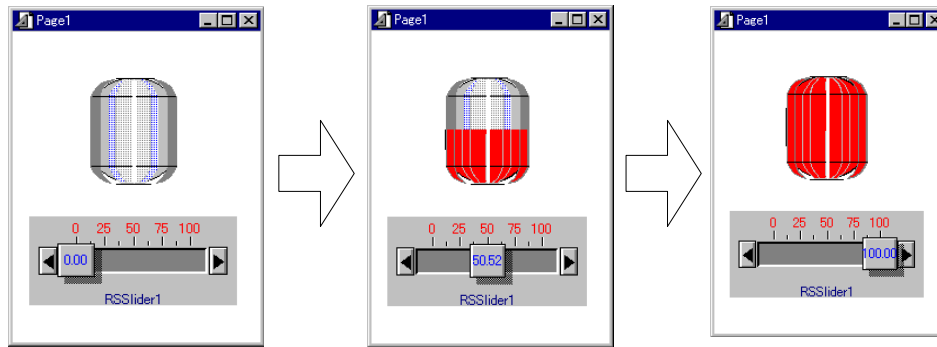
図　3.7.12 アニメーション機能（リベール機能）

■ グラフィックオブジェクトの結合

グラフィックオブジェクトを複数からなる図や絵は，オブジェクトを結合することにより一つのグラフィックオブジェクトとして扱うことができます。これにより，グラフィックオブジェクトの再利用がしやすくなり，作業効率が向上します。
下記に，計器図のグラフィックオブジェクトを結合した場合のサンプルを示します。
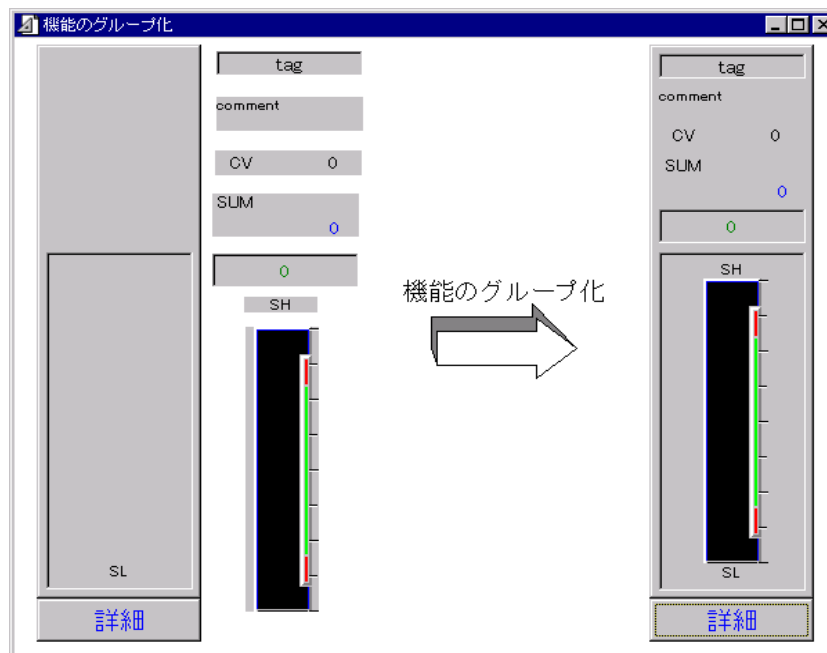
図　3.7.13 グラフィックオブジェクトのグループ化

■ リアルタイムトレンド表示

ASTMAC では以下のようなトレンドを簡単に表示することができます。詳しくは，本書の「3.10 トレンド」を参照してください。
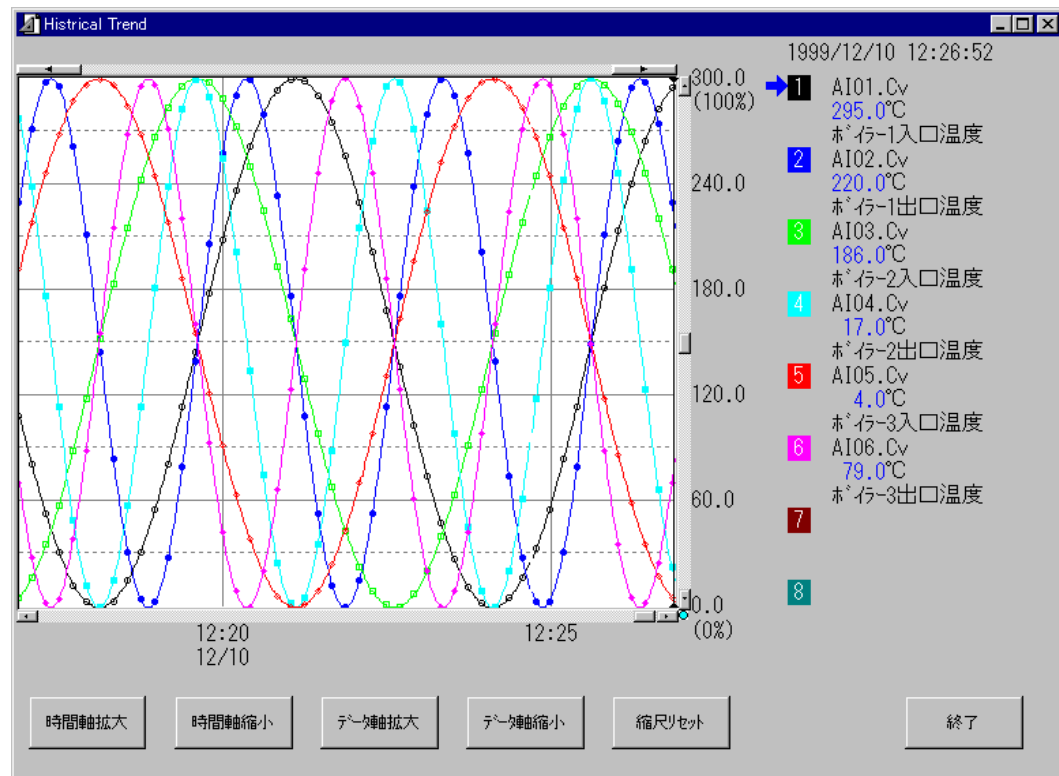
図　3.7.14 トレンドグラフの表示例

■ アラーム表示

ASTMAC では以下のようなアラームを簡単に表示することができます。詳しくは，本書の「3.8 メッセージ管理」を参照ください。

図　3.7.15 アラーム表示の例

■ GKitOCX

GKitOCX のコントロールの種類と表示例を以下に示します。

GKitOCX コントロールの種類

コントロール名称	概要	機能
NT310AJ データ入力 コントロール	文字列，数値などを入力する画面を構成する時に使用する ActiveX コントロールです。	入力形式（大文字，小文字，整数，実数，日付，金額，カスタム） チェック機能（上下限値，最大・最小文字数） エラー処理（異常イベント，音声警告，表示色変更，メッセージボックス）
NT311AJ 表示形式表示 コントロール	表形式の表示を行なう画面を構成するための ActiveX コントロールです。	入力形式（大文字・小文字，整数，実数，日付，金額，時間，パスワード，カスタム，ピクチャ，リストバックス，コンボックス，プッシュボタン，チェックボックス，マルチライン）
NT312AJ グラフ表示 コントロール	大量データのグラフ表示を行なう ActiveX コントロールです。	グラフの種類（折れ線グラフ，円グラフ，棒グラフ，ON/OFF グラフ，散布図）

図　3.7.16 データ入力の例

Page1

	A地区			B地区			
	雨量	水位	使用量	雨量	水位	使用量	雨量
01日	5.0	3.00	0	5.0	3.00	0	0.0
02日	0.0	0.00	0	0.0	0.00	0	0.0
03日	0.0	0.00	0	0.0	0.00	0	0.0
04日	0.0	0.00	0	0.0	0.00	0	0.0
05日	0.0	0.00	0	0.0	0.00	0	0.0
06日	0.0	0.00	0	0.0	0.00	0	0.0
07日	0.0	0.00	0	0.0	0.00	0	0.0
08日	0.0	0.00	0	0.4	0.00	0	0.0
09日	0.0	0.00	0	0.0	0.00	0	0.0
10日	0.0	0.00	0	0.0	0.00	0	0.0
11日	0.0	0.00	0	0.0	0.00	0	0.0
12日	0.0	0.00	0	0.0	0.00	0	0.0
13日	0.0	0.00	0	0.0	0.00	0	0.0
14日	0.0	0.00	0	0.0	0.00	0	0.0
15日	0.0	0.00	0	0.0	0.00	0	0.0
16日	10.0	8.00	0	12.0	10.00	0	0.0
17日	7.0	5.00	0	0.0	0.00	0	0.0
18日	0.0	0.00	0	0.0	0.00	0	0.0
19日	0.0	0.00	0	0.0	0.00	0	0.0
20日	0.0	0.00	0	0.0	0.00	0	0.0
21日	0.0	0.00	0	0.0	0.00	0	0.0
22日	0.0	0.00	0	0.0	0.00	0	0.0
23日	0.0	0.00	0	0.0	0.00	0	0.0
24日	0.0	0.00	0	0.0	0.00	0	0.0
25日	0.0	0.00	0	0.0	0.00	0	0.0
26日	0.0	0.00	0	0.0	0.00	0	0.0
27日	0.0	0.00	0	0.0	0.00	0	0.0
28日	3.0	2.00	0	0.0	0.00	0	0.0
29日	10.0	9.00	0	0.0	0.00	0	0.0
30日	0.0	0.00	0	0.0	0.00	0	0.0
31日	0.0	0.00	0	0.0	0.00	1	0.0
合計	0.0	0.00	0	3.0	0.00	1	0.0
最大	0.0	0.00	0	2.0	0.00	1	0.0
最小	0.0	0.00	0	0.0	0.00	0	0.0
平均	0.0	0.00	0	0.0	0.00	0	0.0

図　3.7.17 表示形式表示の例
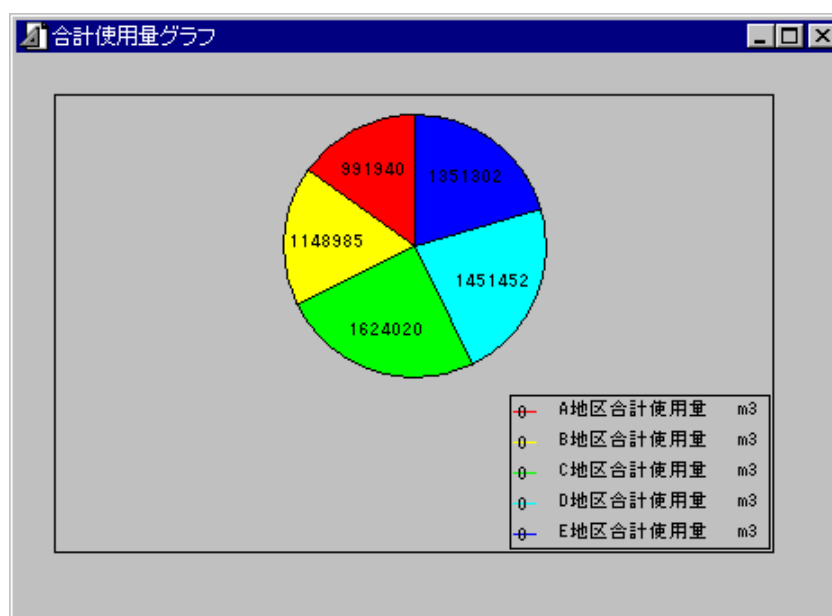
図　3.7.18 円グラフの例
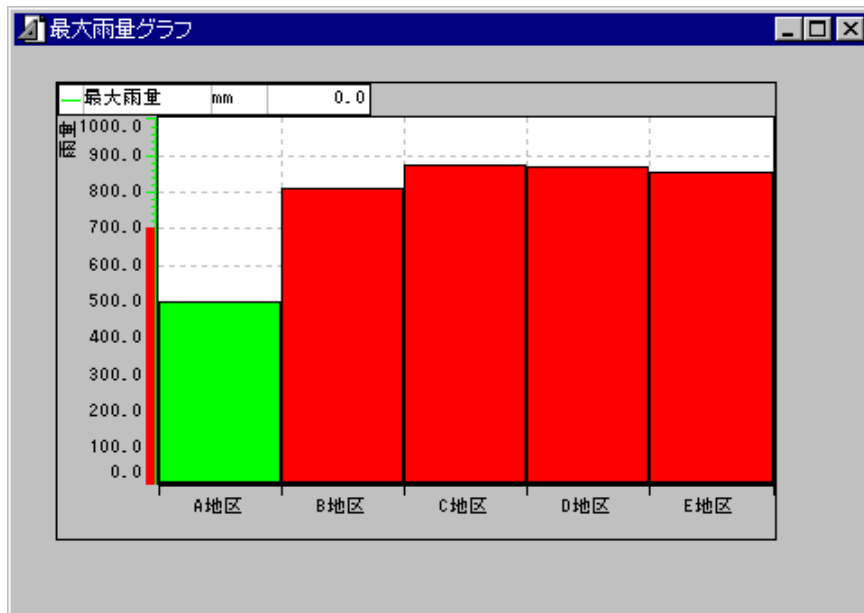
図 3.7.19 棒グラフの例
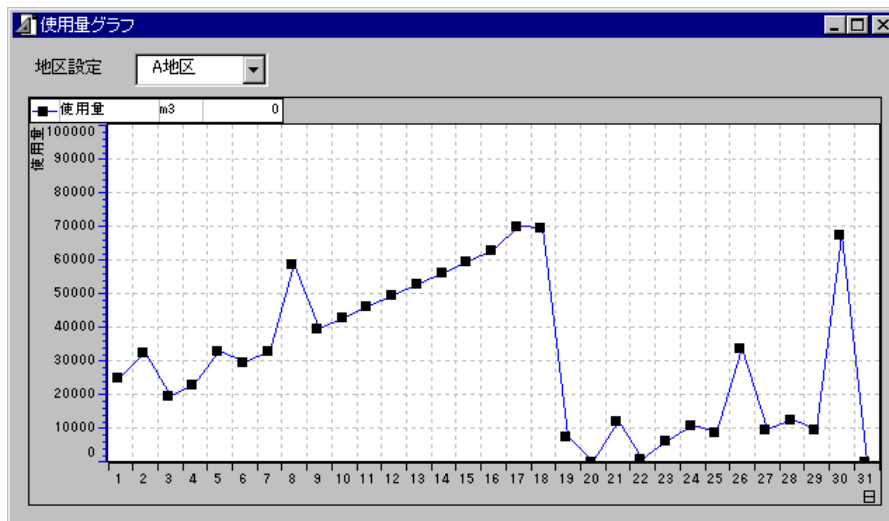
図 3.7.20 折れ線グラフの例
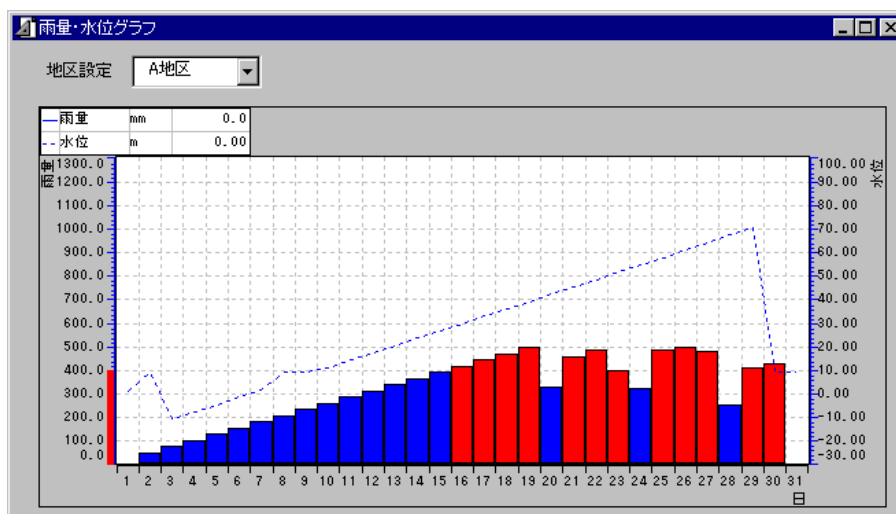
図 3.7.21 オーバレイグラフの例

3.7.3 画面間の関連づけ

ここまでで，システムに必要な各画面の設計は終わりました。
画面はユーザに分かりやすいように画面の関連づけ（展開）を考えなくてはなりません。
ASTMAC には，1 機能 ＝ 1 タスク ＝ 1 プロジェクトという関係があります。
実行ファイルは，プロジェクトごとに作成します。
1 プロジェクトは複数の画面（Page）を持つことができ，プロジェクト内での画面展開を設計します。プロジェクトにまたがるような画面展開は行わないように，タスクの設計を行います。
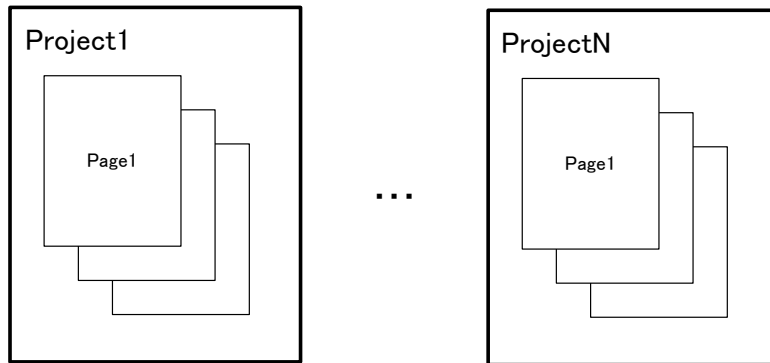
図　3.7.22 プロジェクトと Page の関係

画面間の関連づけを行うには，設計した画面がどのように展開されるかを図に表します。
生産管理 ASTMAC に以下の画面を設計したとします。

- メニュー画面
- 組み立てライン監視画面
- 消費電力監視画面
- 組立完成実績一覧画面
- アラームメッセージ履歴画面

上記画面間の関連付けを行い，画面展開図作成します。
「図 3.7.23 画面展開図」は，画面展開図の例を示したものです。
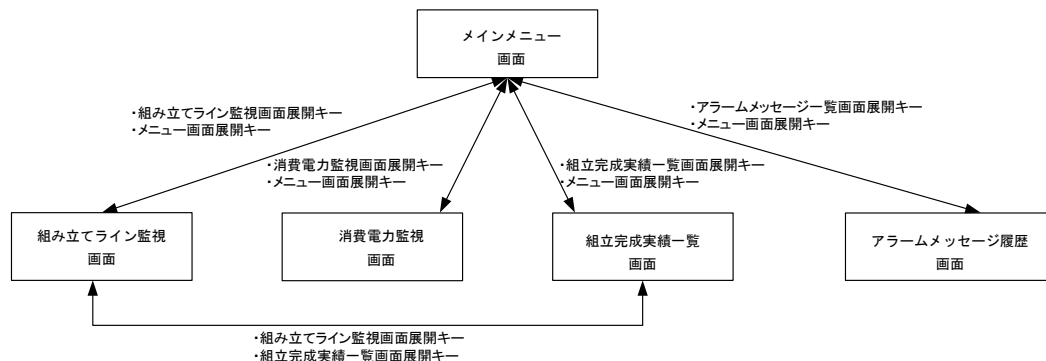
図　3.7.23 画面展開図

3.7.4 画面とコントロールオブジェクトデータの結合

画面とデータとの結合には以下の 2 通りがあります。
- プロパティリンク
- 無手順アクセス

● プロパティリンク

データサーバのコントロールオブジェクトのプロパティ値を，グラフィックオブジェクトの任意のプロパティ値へ自動的に反映する機能をプロパティリンクと呼びます。
プロパティリンクによりデータサーバのオブジェクトとグラフィックのオブジェクトの協調動作を実現することができます。

● 無手順アクセス

無手順アクセスは VBA で行います。
グラフィックビルダの VBA から，データサーバのコントロールオブジェクトに無手順でアクセスすることが可能です。

詳細は，「ASTMAC グラフィック機能解説書（IM 34P02E01-01） 7.データサーバとの接続」を参照してください。

補足

プロパティリンクには，直接指定と間接指定の 2 通りの方法があります。直接指定は，データサーバのオブジェクトのプロパティをグラフィックオブジェクトへ直接ドラッグ＆ドロップして容易に設定することが可能です。固定の I/O オブジェクトとグラフィックオブジェクトをリンクするときに使用します。間接指定は，グラフィックオブジェクトの VBA で記述して使用します。複数の I/O オブジェクトとグラフィックオブジェクトをリンクするときに使用します。

- プロパティリンク名の直接指定
  作成する手順は 3 種類あります
  「グラフィックオブジェクトへ直接にドラッグ＆ドロップする」
  「プロパティリンクウィンドウへドラッグ＆ドロップする」
  「プロパティリンクウィンドウの内側を右クリックする」

- プロパティリンク名の間接指定
  プロパティリンク名を定義するとき，完全なオブジェクト名称でなく，一部分を，｛｝に囲まれた文字列で間接的に指定することができます。

  例
  `ASTMAC.{Machine}.{Group}.AI01.Cv`　（`{Machine},{Group}`が間接指定）

  プロパティリンク名の間接指定を使うと，VBA のプログラムからダイナミックにプロパティリンクを切り替えることができるので，Page の再利用性があがる，ダイナミックに切り替える際の VBA プログラミングが容易であるといったメリットがあります。詳細は，「ASTMAC グラフィック機能解説書（IM 34P02E01-01） 9.便利な機能の紹介」を参照してください。

3.7.5　プログラム間のデータ共有

プログラム間のデータ共有を行なうのに有効なオブジェクトが「ブロックデータ」と「シグナル交信」です。
ブロックデータオブジェクトにより，タスク間のデータ共有を簡単に行うことができます。ブロックデータオブジェクトは，データサーバ内に最大 32Kbyte までの共有データを保持することができます。このブロックデータオブジェクトはデータサーバ上に複数貼り付けることができます。この共有データはアプリケーションフォームからブロックデータアクセスオブジェクトを使用してアクセスすることができます。
また，シグナル交信オブジェクトはデータサーバ，アプリケーションフォームのいずれのコンテナにも配置することが可能で，プロセス間で事象の発生（シグナル）の送信および受信を行います。

下図は，ブロックデータオブジェクトとシグナル交信を使って，2 つのアプリケーションフォームでデータの受け渡しにブロックデータオブジェクトとシグナルオブジェクトを使った例です。
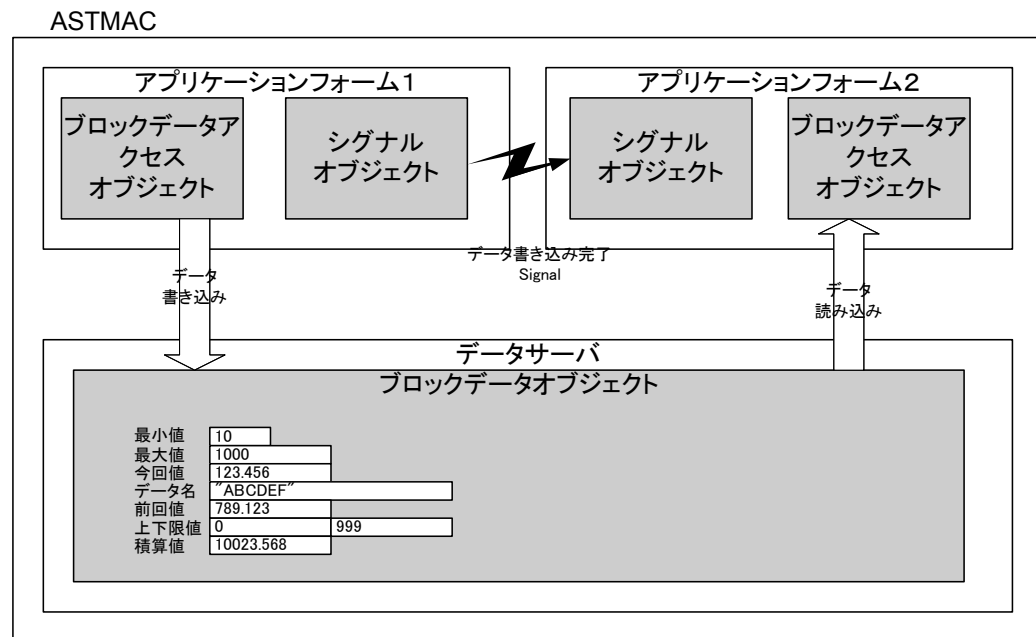
図　3.7.24 プログラム間のデータ共有

3.7.6 I/O機器とのデータ送受信

ブロックデータオブジェクトは，I/O 機器との大量データ送受信を行う，一種のバッファとしての機能を持ちます。ブロックデータオブジェクトが保持する全データ，または一部を一括して I/O 機器にライトすることや I/O 機器からリードすることができます。

下図は，I/O 機器から読み込んだデータをブロックオブジェクト内の共有データに書き込み，ブロックデータアクセスオブジェクトを用いてアプリケーションフォームからアクセスしています。また，アクセスのタイミングをとるのにシグナル交信を使っています。

詳細は，「ASTMAC マルチタスク支援パッケージ説明書（IM 34P02H04-01） 2.3 I/O 機器との大量データの送受信」を参照してください。
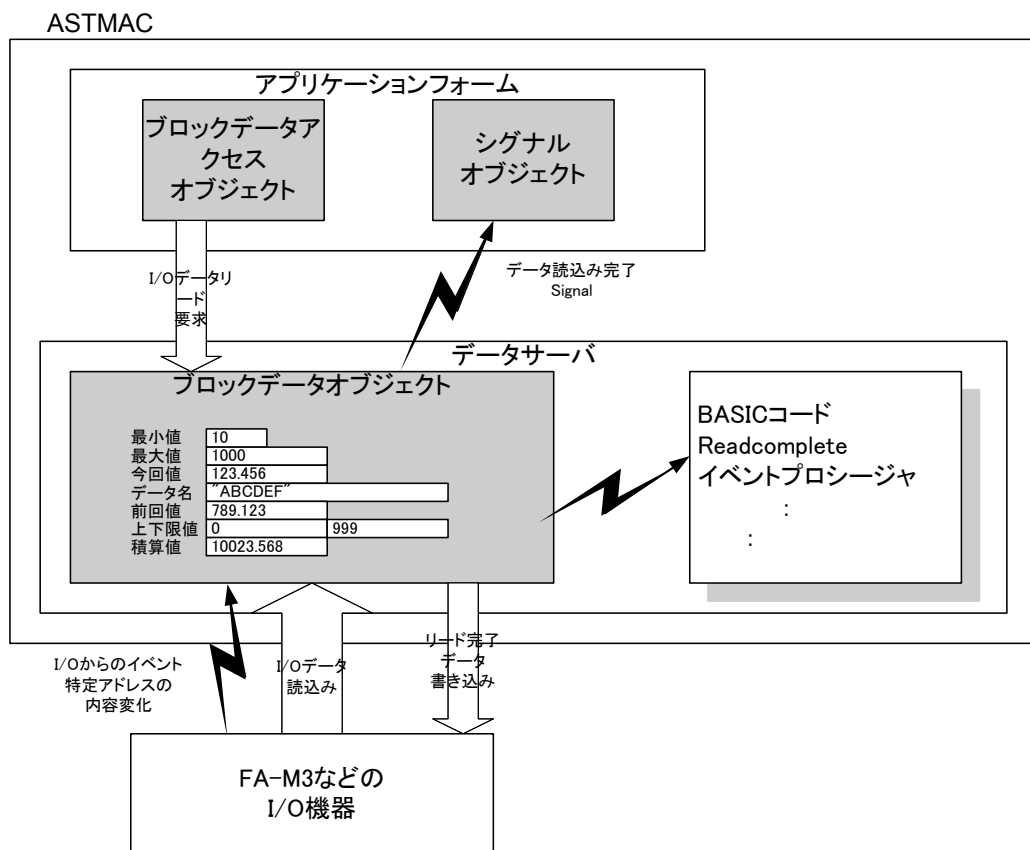
図 3.7.25 I/O 機器とのデータ送受信

3.8 メッセージ管理

システムのアラームや操業に関わるメッセージをオペレータに知らせ，オペレータがシステムの状況を把握できるようにすることは，ASTMAC で構築するソフトウェアにとって非常に重要な機能となります。このようなメッセージを統括管理し，オペレータに通知してくれる機能が，ASTMAC のメッセージ管理機能です。
ASTMAC のメッセージ管理機能の概要を以下の図で示します。
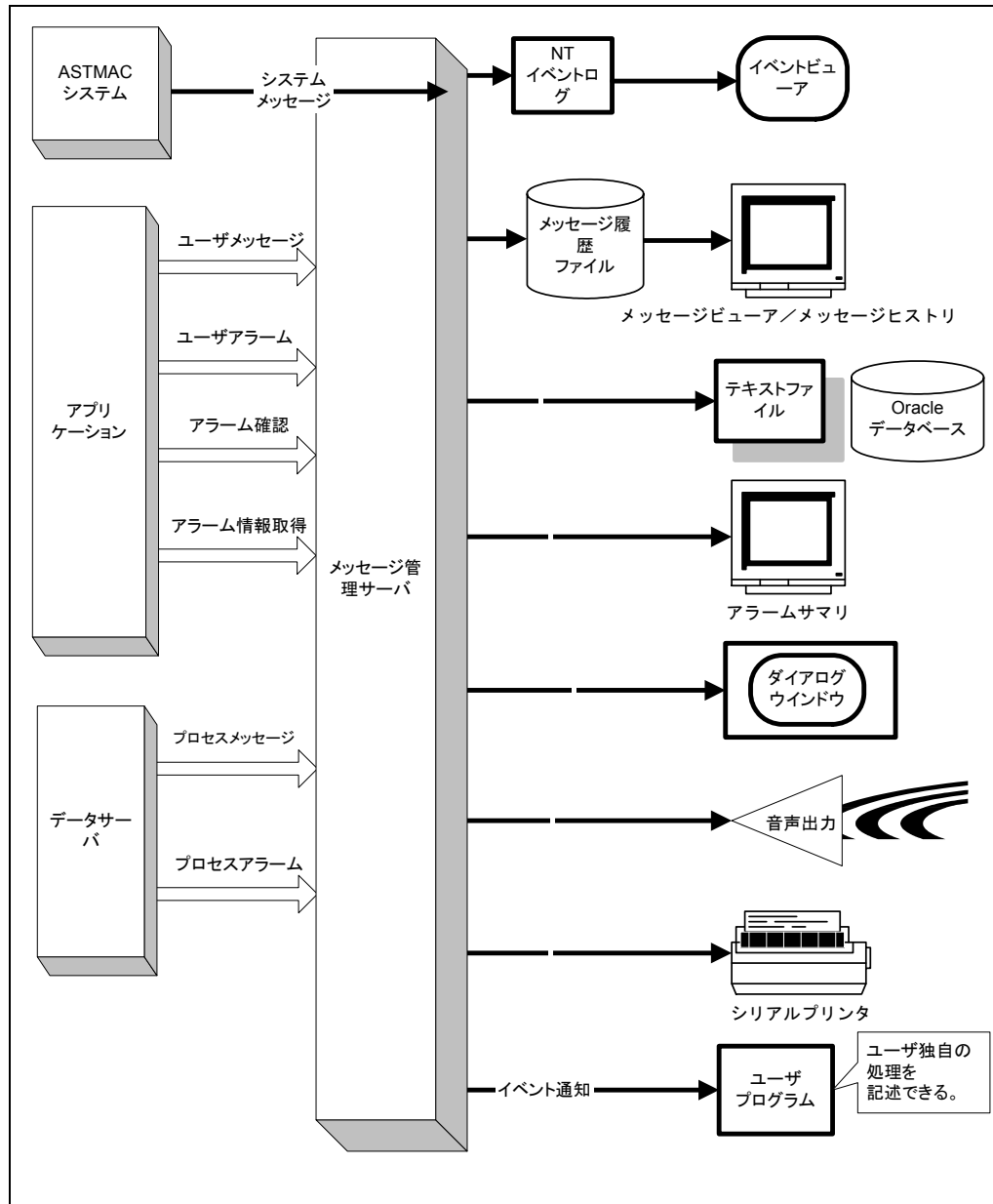
図　3.8.1 メッセージ管理の位置付け

メッセージ管理は，基本システムやアプリケーションプログラムで発生するエラー，アラーム，操作履歴，操業履歴などのメッセージを統括的に管理するものです。基本システムだけではなく，アプリケーションシステムを含めたシステム全体の操業，保守を支援することを目的としています。これにより，ASTMAC の基本システムやユーザアプリケーションで発生したメッセージをオペレータに通知することができます。

ASTMAC のメッセージは「表 3.1 ASTMAC メッセージタイプ一覧」に示す分類があります。

表　3.1 ASTMAC メッセージタイプ一覧

メッセージタイプ		内容
システムメッセージ		ASTMAC システムプログラムからのメッセージです。Windows イベントログへ保存されます。
アプリケーションメッセージ	ユーザメッセージ	アプリケーションフォームから出力されるメッセージで，オペレータの確認操作を必要としないものです。
	ユーザアラーム	アプリケーションフォームから出力されるメッセージで，オペレータの確認操作を必要とするものです。
コントロールオブジェクトメッセージ	プロセスメッセージ	データサーバのコントロールオブジェクトが出力するメッセージで，オペレータの確認操作を必要としないものです。
	プロセスアラーム	データサーバのコントロールオブジェクトが出力するメッセージで，オペレータの確認操作を必要とするものです。

● システムメッセージ

ASTMAC システムプログラム（システムの基本部分）から出力するエラーメッセージや情報メッセージです。これらのメッセージは，メッセージ管理経由で WindowsNT におけるイベントビューア内のアプリケーションログで確認することができます。

● アプリケーションメッセージ

アプリケーションプログラムが発生する操作履歴や一般情報です。特にオペレータによる確認操作が必要なメッセージを「ユーザアラーム」と呼び，オペレータによる確認操作が必要でないものを「ユーザメッセージ」と呼びます。アプリケーションメッセージは，メッセージビルダによって定義を行います。そして，アプリケーションフォームおよびデータサーバの VBA でメッセージ送信命令を実行することによって発生させます。

● コントロールオブジェクトメッセージ

コントロールオブジェクト（主にデバイスタグオブジェクト）に関するメッセージです。オペレータによる確認操作が必要なメッセージを「プロセスアラーム」と呼び，オペレータによる確認操作が必要でないものを「プロセスメッセージ」と呼びます。コントロールオブジェクトメッセージはオブジェクトビルダにて定義します。発生条件とメッセージはコントロールオブジェクト毎に決まっており，オブジェクト毎に発生の有効／無効と発生時の処理を指定します。

アプリケーションメッセージとコントロールオブジェクトメッセージには，オペレータの確認が必要な「アラーム」とオペレータの確認操作が必要でない「メッセージ」の 2 種類が存在します。

● アラーム

温度異常や装置の重故障など，オペレータが発生したことを気付かずにいると重大な事故になりかねないようなメッセージは「アラーム」として出力します。「アラーム」はオペレータの確認操作を必要とし，アラームサマリにおいて未確認のアラームが一覧表示できるようになっています。

● メッセージ

バッチ開始や装置の起動など，オペレータが気付かずにいても事故が発生しないメッセージは，単純な「メッセージ」として出力します。「メッセージ」はオペレータの確認操作の必要がなく，オペレータへ状態変化を通知したり，操業履歴や操作履歴として記録することを目的とします。

各種メッセージは「表 3.2 メッセージの属性」に示す属性を持っています。

表　3.2 メッセージの属性

属性	説明
タイプ	システムメッセージ，ユーザメッセージ，ユーザアラーム，プロセスメッセージ，プロセスアラームに分類されます。
ソース	メッセージ発生元を示します。プログラム名，機能名，工程名など自由に設定できます。
番号	メッセージに割り振られた番号です。ソース毎に 1 から 9999 までの番号を使えます。
カテゴリ	メッセージのレベル，意味を表します。
テキスト	メッセージそのものです。
出力処理	メッセージ出力先を指定するものです。メッセージ履歴ファイル出力，テキストファイル出力，ダイアログウィンドウ表示，音声出力，プリンタ印字です。但し，プロセスメッセージはメッセージ履歴ファイル出力，テキストファイル出力のみ可能です。

次にタイプ毎の属性の詳細を示します。

表　3.3 タイプごとの属性詳細

タイプ	ソース	番号	カテゴリ
システムメッセージ	システムプログラム名 先頭に”YFS”が必ず付く	1～9999	障害(1) 警告(2) 情報(3)
ユーザメッセージ	任意	ソース毎に 1～9999	障害(1)，警告(2)，情報(3)，デバッグ(0)が標準で用意されており，その他上記を含めて最大 16 まで定義可能
ユーザアラーム	上記と同じ	上記と同じ	上記と同じだが，デバッグ(0)は使用不可
プロセスメッセージ	デフォルトで AREAn”が用意されており，オブジェクトビルダで名前の変更が可能	システム予約	情報(0)のみ
プロセスアラーム	上記と同じ	上記と同じ	1～15 オブジェクトビルダで設定した重要度

基本設計の段階において，オペレータが必要とするメッセージには何があるかを検討し，メッセージ一覧表を作成します。このメッセージ一覧表に記載されたメッセージごとに「表 3.2 メッセージの属性」で示した属性をすべて付与することにより，メッセージに関する設計を行います。

● メッセージのタイプ属性について

まず，一覧表に示されたメッセージがアプリケーションメッセージなのか，コントロールオブジェクトメッセージなのかを判断する必要があります。
1 つのコントロールオブジェクトの状態変化に伴い発生するメッセージであれば，メッセージのタイプ属性をプロセスアラーム，プロセスメッセージとします。また，コントロールオブジェクトの状態変化に関係しないメッセージや，複数のコントロールオブジェクトの状態変化に関わるメッセージであれば，メッセージのタイプ属性をユーザアラーム，ユーザメッセージとします。

タイプ属性については上記の通り判断し，その他の属性についてはタイプ属性ごとに各々の属性を検討します。

● アプリケーションメッセージの検討

① タイプ
システムに障害を与える恐れのあるものはユーザアラームとし，障害を与える恐れの無いものをユーザメッセージとします。操作履歴，操業履歴などはシステムに障害を与える恐れがないため，すべてユーザメッセージとして取り扱います。

② ソース
メッセージ発生元のアプリケーションフォームの機能名称をソース名とします。
例えば，「生産指示管理」，「上位通信」，「組立実績管理」などの名称を用います。

③ 番号
ソース名毎に 1～9999 の番号を使用することができますが，ユーザアラームを 1 から 999 番まで，ユーザメッセージを 1001 から 1999 番までと規定すると，メッセージの混乱を避けることができます。

④ カテゴリ
カテゴリには障害（1），警告（2），情報（3）を使用します。
ユーザアラームはシステムに対する警告であり，基本的には「警告（2）」を使用し，既にシステムの障害であるという場合に「障害（1）」を使用するようにします。また，ユーザメッセージは「情報（3）」のみ使用し，アラームとメッセージを根本的に切り分けます。
カテゴリの例を示します。
　障害：上位通信異常，ファイルアクセスエラー，など
　警告：搬送ライン満杯，生産指示未登録，など
　情報：生産指示情報受信，生産実績送信，など
なお，ユーザメッセージには「デバッグ（0）」を使用することができるようになっていますが，あくまでもシステムを本稼動へ移行させるまでの試験期間でのみ使用するものであって，本稼動で発生させるべきメッセージではありません。デバッグメッセージについては後で解説します。
また，カテゴリにはこれら以外にも登録することができますが，カテゴリを増やすことはメッセージの重要度を複雑にすることになりますので，これら以外のカテゴリを登録することは避けるべきです。カテゴリはここで示した 3 種（デバッグを含め 4 種）までで使用することをお勧めします。

⑤ テキスト
出力するメッセージを指定します。また，メッセージ中に最大 6 個までの可変情報を含めることもできます。例えば，「生産指示情報を 6 件受信しました。」のように任意の受信件数を出力することも可能となります。

⑥ 出力処理
出力処理はユーザアラーム，ユーザメッセージとも，メッセージ履歴ファイル出力，テキストログファイル出力，ダイアログウィンドウ表示，音声出力，プリンタ印字が可能です。すべてのアプリケーションメッセージはメッセージ履歴ファイル出力，テキストログファイル出力を行うようにしておき，その他の出力は必要に応じて設定するようにします。

● コントロールオブジェクトメッセージの検討

① タイプ
システムに障害を与える恐れのあるものはプロセスアラームとし，障害を与える恐れの無いものをプロセスメッセージとします。プロセスアラームは DI／AI／XAI のデバイスタグオブジェクトのみ出力可能となります。また，プロセスメッセージは DI オブジェクトと I/O 機器への出力を伴うデバイスタグオブジェクトについてのみ出力可能です。

② ソース
デフォルトで用意されている AREA1～AREA16 を工程名称に置き換えて使用し，メッセージ発生元の工程名をソース名として使用します。

③ 番号
コントロールオブジェクトメッセージの場合，番号は固定です。
変更することはできません。

④ カテゴリ
プロセスアラームの場合はカテゴリを重要度として取り扱います。重要度は 1 から 15 まで設定可能ですが，アプリケーションメッセージの場合と同様に，障害，警告の 2 種類まででで使用するようにします。障害を重要度 1 として取り扱い，警告を重要度 2 として取り扱うようにします。
また，プロセスメッセージはカテゴリの選択ができません。情報固定となります。

⑤ テキスト
プロセスアラーム，プロセスメッセージとも出力されるテキストの内容は固定で，アプリケーションにて変更することはできません。出力されるテキストの内容を以下に示します。

表　3.4 プロセスメッセージ一覧

番号	メッセージ内容	備考
1000	タグ名 NR CV=Val タグコメント	出力時のメッセージ
1000	タグ名 COS CV=Val タグコメント	状態変化発生 “COS”文字列はユーザが変更可能
1000	タグ名 CFN CV=Val タグコメント	正常状態からの変化

CV=Val の Val は現在値を表す文字列です。

表 3.5 プロセスアラーム一覧

番号	メッセージ内容	備考
1	タグ名 ACK [確認者 ID] タグコメント	確認操作履歴
1000	タグ名 NR CV=Val タグコメント	“正常復帰”時のメッセージ
1010	タグ名 COMM CV=Val タグコメント	通信エラー
1020	タグ名 IOP CV=Val タグコメント Priority=N	入力オープン
1030	タグ名 COS CV=Val タグコメント Priority=N	状態変化発生 “COS”文字列はユーザが変更可能
1040	タグ名 CFN CV=Val タグコメント Priority=N	正常状態からの変化
1050	タグ名 HH CV=Val タグコメント Priority=N	上上限異常
1060	タグ名 LL CV=Val タグコメント Priority=N	下下限異常
1070	タグ名 H CV=Val タグコメント Priority=N	上限異常
1080	タグ名 L CV=Val タグコメント Priority=N	下限異常
1090	タグ名 VEL CV=Val タグコメント Priority=N	変化率異常
1100	タグ名 HISUM CV=Val タグコメント Priority=N	積算オーバー
1110	タグ名 HIRSUM CV=Val タグコメント Priority=N	入力時間積算オーバー
1120	タグ名 HIESUM CV=Val タグコメント Priority=N	入力回数積算オーバー

CV=Val の Val は現在値を表す文字列です。
また，Priority=N の N は 1 から 15 までの重要度です。

⑥ 出力処理
プロセスアラームの場合は，メッセージ履歴ファイル出力，テキストファイル出力，ダイアログウィンドウ表示，音声出力，プリンタ印字が可能ですが，プロセスメッセージの場合は，テキストファイル出力，ダイアログウィンドウ表示のみ可能となります。

● メッセージの出力方法

コントロールオブジェクトメッセージはオブジェクトビルダで登録したアラームプロパティに従い ASTMAC が自動で出力するため，ユーザが任意に出力することはできませんが，アプリケーションメッセージはユーザが任意のタイミングで出力することができます。
アプリケーションメッセージは，メッセージビルダで登録した内容を，アプリケーションインタフェースを用いて出力する方法を取ります。インタフェースの詳細については，「ASTMAC エンジニアリングマニュアル（IM 34P02D01-01） PART-C 2-1 メッセージ送受信インタフェース」を参照してください。なお，メッセージビルダで登録していないメッセージを出力するインタフェースも用意されていますが，出力するメッセージを管理する上で混乱を招きますので，原則として，メッセージビルダで登録した内容を出力する方法を使用します。

● メッセージの出力先

メッセージはメッセージが発生したタイミングで，個々のメッセージの属性で指定された出力先に出力されます。

① メッセージ履歴ファイル出力
メッセージ履歴ファイルへ出力された内容は，ASTMAC のメッセージビューア，メッセージヒストリ（MsgHIstX コントロール）で確認することができます。
メッセージ履歴ファイルの保存件数は最大 5000 件で 5000 件を超えた場合は，最古のメッセージが削除されます。

② テキストログファイル出力
日単位に作成されるテキストファイルに発生したメッセージが発生順に格納されており，Windows のメモ帳で内容を確認することができます。ファイルは 30 日間保存され，30 日を超えるものは自動的に削除されます。テキストログファイルのファイル名は以下の通りで，保存先はメッセージ管理環境設定で指定したディレクトリとなります。

logyyyymmdd.txt
（ 例:log20100401.txt [2010 年 4 月 1 日分のメッセージログ] ）

③ ダイアログウィンドウ表示（プロセスメッセージは不可）
発生メッセージをダイアログ表示します。ダイアログ表示中は新たなメッセージが発生しても新規にダイアログを表示することはなく，ダイアログ表示キューに溜められます。
オペレータがダイアログ表示を消去することにより，ダイアログ表示キューに格納されているダイアログが表示されます。また，ダイアログ表示キューに格納可能なメッセージ数は，メッセージ管理環境設定で指定した数となり，格納可能数を超えて発生したメッセージに対してはダイアログが表示されません。

④ 音声出力（プロセスメッセージは出力不可）
Windows の Wave ファイルを指定することによりアラーム音を鳴らすことができます。また，Wave ファイルに音声を登録することにより，ボイスアナンを出力することもできます。アラーム音はオペレータがアラーム確認をするか，アプリケーションプログラムからアラーム音を停止させることによって止めることができます。
また，発生したメッセージを外部データベースに書き込むこともできます。
RDBMS は Oracle8.05 に限り，Oracle Object for OLE（OO4O）を使用してメッセージ管理が自動的に書き込むようになります。Oracle 側の表の仕様や，データベースを使用する場合の設定方法などについては，「ASTMAC エンジニアリングマニュアル（IM 34P02D01-01）」を参照してください。

● メッセージの表示，オペレータの確認

出力されたプロセスアラーム，プロセスメッセージ，ユーザアラーム，ユーザメッセージは ASTMAC のメッセージヒストリコントロール，または，メッセージビューアで確認することができます。また，プロセスアラームとユーザアラームについては，アプリケーションフォームにアラームサマリ用 ActiveX （AlarmFsX）を貼り付けることで，アラーム情報を一覧表示することができます。
アラームサマリには表示制御用の各種プロパティがあり，表示をユーザがカスタマイズできるようになっています。詳しくは「ASTMAC エンジニアリングマニュアル（IM 34P02D01-01）」を参照してください。
オペレータによるアラームの確認操作は，AlarmFsX コントロール内に表示されている発生中アラームをダブルクリックすることにより行います。アプリケーションフォームの VBA からアラーム確認を行う場合は，AlarmFsX コントロールのメソッドを用いる方法があります。また，ActiveX オートメーションインタフェースを用い，VBA からアラーム確認を行うこともできます。

● デバッグメッセージについて

アプリケーションメッセージにはデバッグのカテゴリが用意されており，ユーザが試運転期間中にシステムの動作を監視するために出力することができるようになっています。
ユーザメッセージのみデバッグのカテゴリが指定可能で，ユーザアラームに指定するとメッセージが正しく出力されなくなります。また，デバッグメッセージの出力は必ずテキストログファイルのみとなり，メッセージ履歴ファイルやダイアログ出力，プリンタ印字は一切できません。メッセージログはメッセージビューアでデバッグログを指定すると，テキストログファイルに保存されるようになります。

● アプリケーションプログラムでメッセージを捕獲する

ユーザが作成したアプリケーションフォーム VBA やデータサーバにて，発生したメッセージを捕獲することができます。この機能を使うことでコントロールオブジェクトメッセージをアプリケーションプログラムで取り扱うことができます。つまり，オブジェクトビルダやメッセージビルダでメッセージを定義する際にアプリケーション通知先を設定することにより，指定されたアプリケーションフォームが該当するメッセージを受信することができるようになります。詳しくは「ASTMAC エンジニアリングマニュアル（IM 34P02D01-01）PART-C 2.メッセージ送受信インタフェース」を参照してください。

3.9　帳票設計

生産実績やユーティリティ使用量実績，または管理日報などを記録として残し，監督部署に報告することはシステムを運用する上で非常に重要な機能となります。自動化されたシステムではこの報告書を，アプリケーションプログラムにて，帳票として自動出力することが一般的です。この帳票の作成をプログラミングレスで行えるのが ASTMAC のレポート管理機能です。レポート管理機能は ASTMAC の帳票パッケージのライセンスを登録することにより，使用することができるようにます。

3.9.1　レポート管理機能の概要

レポート管理機能の概要を「図 3.9.1 レポート管理機能概要」に示します。

図　3.9.1 レポート管理機能概要

① 帳票ビルダ
帳票の各種定義を行うビルダです。
最大 16 種類の収集ブロックを定義することができます。各収集ブロックには，最大 32 点の収集データ，帳票の種別，締め切り時刻と締め切り集計処理，自動印字の有無，収集条件，RDB の指定などを行います。収集ブロックを定義することにより，ブロックごとに，該当する印刷用テンプレートとサポートプログラム（印刷用マクロ）が作成され，1 枚の帳票として出力できるようになります。

② 帳票定義情報
帳票ビルダで登録した収集ブロックの情報などを格納するためのファイルです。
この定義情報をもとにレポート管理がデータの収集や締め切り処理を行います。

③ レポート管理
帳票ビルダで登録された帳票定義情報にもとに，運転時に帳票作成を実行するプログラムです。データサーバから定周期にデータを収集し，ISAM ファイルに記録します。さらに，締め切り時刻には締め切りデータを演算後，ISAM ファイルに記録した後，Excel サポートプログラムを起動します。

④ 帳票データ用 ISAM
帳票ビルダで定義した収集ブロック単位に必要な ISAM ファイルが作成され，レポート管理によって，定周期にデータが保存されます。また，締め切りデータも同じファイル内でっ管理されます。

⑤ 外部 RDB（Relational DataBase）
帳票ビルダで定義した内容に従い外部データベースへの帳票データの書き込みを行うことができます。

⑥ Excel サポートプログラム（印刷用マクロ）
帳票ビルダで定義した帳票ブロック毎に Excel サポートプログラム（印刷用マクロ）が生成されます。Excel サポートプログラムはレポート管理，もしくはアプリケーションから起動され，ユーザが作成した背景データとマスターブックに設定した内容に従い，帳票を印字します。

⑦ 印刷用テンプレート
帳票ビルダで定義した帳票ブロック毎に作成され，ユーザが帳票のフォーマットを定義するための Microsoft Excel 用テンプレートファイルです。帳票のタイトルや，罫線などを記述しておくことにより，自動的にデータが埋め込まれ，帳票として出力されます。

⑧ Excel マスターブック
Excel サポートプログラムが帳票を印刷するために必要な情報を，帳票ブロック単位に定義するためのファイルです。定周期データの印字開始位置や，締め切りデータの印字開始位置，などを定義することにより，帳票にデータを打ち出すことが可能となります。

以上のように，ASTMAC は Microsoft 社の Excel（注）がインストールされていることを前提にレポート管理機能が構築されています。帳票パッケージを使用したアプリケーションを作成する場合は，ASTMAC の帳票パッケージをインストールした PC に，Excel（注）をインストールしておく必要があります。

注： Excel のバージョンは製品仕様を参照ください。

3.9.2 ASTMACがサポートする帳票の種類

ASTMAC から出力可能な帳票は日報，月報，ロット報です。
帳票ビルダで帳票の設計する場合に，該当する帳票がどの様なものであるかを「帳票タイプ」として指定します。「帳票タイプ」には，「日報」，「月報」，「ロット報」，「簡易バッチ報」の 4 タイプにあわせ，日報と月報を同時に処理可能な「日報・月報」が存在し，タイプごとに出力される帳票の内容が異なります。

● **日報**

収集周期ごとにデータを収集し，記録周期ごとに帳票データを演算し，演算結果を帳票データ用の ISAM ファイルに格納します。また指定された締め切り時刻に，記録周期ごとの演算結果をもとに締め切り演算処理を行い，その演算結果を帳票データ用の ISAM ファイルに格納します。これらの記録結果をもとに，締め切り処理後に日報が出力されます。

● **月報**

収集周期ごとにデータを収集し，1 日単位に帳票データを演算し，演算結果を帳票データ用の ISAM ファイルに格納します。また，毎月 1 日の指定された締切時刻に，1 日ごとの演算結果をもとに締め切り演算処理を行い，その演算結果を帳票データ用の ISAM ファイルに格納します。これらの記録結果をもとに，締め切り処理後に月報が出力されます。

● **日報・月報**

帳票のタイプを日報と指定した場合と同じ処理を行い日報を作成します。
月報は，毎月 1 日の日報締め切り処理後，日単位の締め切り演算で得た結果をもとに演算された結果が月締めのデータとなります。

● **ロット報**

トリガとなる事象が発生してから終了するまでの間，収集周期ごとにデータを収集し，記録周期ごとに帳票データを演算し，演算結果を帳票データ用の ISAM ファイルに格納します。また，締め切り処理はトリガとなる事象が終了したタイミングで行われ，記録された演算結果をもとに演算処理が行われ，その演算結果を帳票データ用の ISAM ファイルに格納します。これらの記録結果をもとに，締め切り処理後の帳票が出力されます。トリガはユーザが VBA から任意に指定可能なイベント型と，デバイスタグオブジェクトの状態変化をもとにするポーリング型があります。

● **簡易バッチ報**

記録開始から終了までの任意の期間内で発生したバッチデータを 1 つの記録報とします。
記録するデータは，バッチ開始および終了時の時刻と，バッチ終了時の瞬時値もしくはバッチ開始時から終了時の差分データです。

3.9.3 帳票作成までの手順

印刷する帳票を設計し，動作確認するまでの手順を以下に示します。

1. 帳票の設計
   ASTMAC から出力する帳票の仕様を，エンドユーザと打ち合わせて取りまとめ，仕様書に記載します。次ページに帳票仕様を取りまとめるためのチェックシートを載せていますので，エンドユーザとの打ち合わせにおいて活用してください。また，本チェックシートは帳票ビルダで設定すべき内容を網羅していますので，帳票ビルダにおける登録操作が大変楽になります。
2. 帳票ビルダの操作
   オブジェクトビルダから帳票ビルダを起動し，エンドユーザとの打ち合わせにおいて活用したチェックシートの内容を登録していきます。
3. 背景部の作成，Master ブックの設定
   背景部を Excel のテンプレートファイルに作成します。ほとんどの場合は既設のシステムにおいて同じような帳票が既に存在しています。そのフォーマットを参考に背景部を作成すればいいでしょう。Master ブックは前景データの印字制御などの情報を記述します。「ASTMAC 帳票パッケージ説明書（IM 34P02H01-01）」を参考に必要なデータを設定します。
4. 印刷テスト
   帳票パッケージのレポート管理機能は操業モードでないと機能しません。
   レポート管理タスクをプロセスコンフィギュレータで登録し，操業モードで立ち上げます。その後，帳票を印刷させ，データの記録，締め切り処理等に問題がないことを確認し，帳票の作成が完了したことになります。

帳票定義シート

ﾌﾞﾛｯｸ名		帳票名	

帳票ﾀｲﾌﾟ	□日報	□月報	□日報・月報	□ロット報
	[____]時締め	[____]時締め	[____]時締め	□ ｲﾍﾞﾝﾄ
		[____]日締め	[____]日締め	□ ﾎﾟｰﾘﾝｸﾞ
	印刷 [___]時[___]分遅れ	印刷 [___]時[___]分遅れ	印刷 [___]時[___]分遅れ	ﾎﾟｰﾘﾝｸﾞ先ﾃﾞﾊﾞｲｽ [____________]

収集周期 記録周期	収集	記録	収集	記録	収集	記録	収集	記録
	□ 1分	□ 5分	□ 1分	□ 5分	□ 1分	□ 5分	□ 1分	□ 5分
	□ 2	□ 10	□ 2	□ 10	□ 2	□ 10	□ 2	□ 10
	□ 5	□ 20	□ 5	□ 20	□ 5	□ 20	□ 5	□ 20
	□ 10	□ 30	□ 10	□ 30	□ 10	□ 30	□ 10	□ 30
	□ 30	□ 60	□ 30	□ 60	□ 30	□ 60	□ 30	□ 60
	□ 60	□ 120	□ 60	□ 120	□ 60	□ 120	□ 60	□ 120
	注）記録周期は必ず収集周期の整数倍になるよう選択すること							

収集ﾃﾞｰﾀ仕様	No.	収集ﾀｸﾞ名	ISAM 列名	表示桁	Index	集計	締切
	1						
	2						
	3						
	4						
	5						
	6						
	7						
	8						
	9						
	10						
	11						
	12						
	13						
	14						
	15						
	16						
	17						
	18						
	19						
	20						
	21						
	22						
	23						
	24						
	25						
	26						
	27						
	28						
	29						
	30						

集計欄：以下の番号で記入(複数選択不可)
- 1－平均値
- 2－最小値
- 3－最大値
- 4－合計値
- 5－差分合計１
- 6－差分合計２
- 7－集計ステータス
- 8－瞬時値

締切欄：以下の番号で記入(複数選択可)
- 1－平均値
- 2－最大値
- 3－最小値
- 4－合計値
- 5－差分合計値
- 6－瞬時平均値
- 7－瞬時最大値
- 8－瞬時最小値
- 9－瞬時最大値時刻
- 10－瞬時最小値時刻
- 11－正常収集率
- 12－（空白）

3.9.4　設定情報の詳細

ここでは，帳票ブロックを設計する上で重要となる項目について説明します。

● 収集周期と記録周期について

収集周期ごとにデータを収集し，記録周期ごとに帳票データとして ISAM ファイルなどに格納します。以下に収集周期と記録周期の関係について示します。
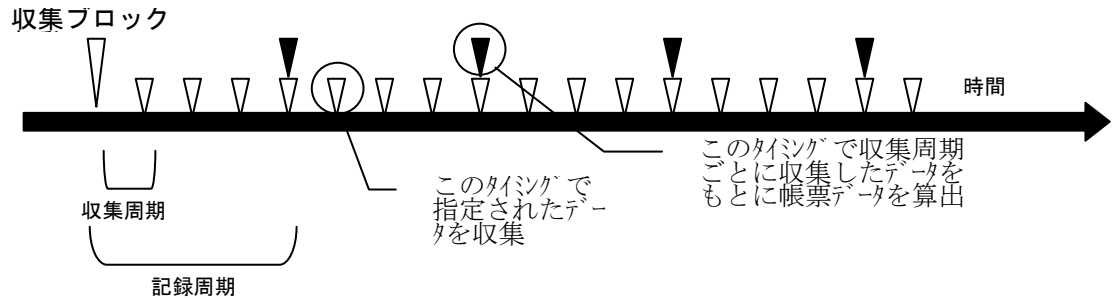
図　3.9.2 収集周期と記録周期

収集ブロックが起動されたタイミングで初回のデータ収集が行われ，以降収集周期ごとにデータ収集が行われます。データ収集後，そのタイミングが記録周期と同じであれば，記録周期内に収集されたデータの集計を行い，データを ISAM ファイルなどに格納します。そのため，収集周期と記録周期は以下のような関係が必要となります。

収集周期≦記録周期
収集周期×N＝記録周期　（N は自然数）

● 記録周期ごとの集計機能について

記録周期ごとに集計機能が実行されます。集計機能には以下のようなものがあります。

収集形式	集計演算の種類		演算規則 補足事項
実測値収集	平均値	（Σ瞬時値）/正常収集回数	無効な瞬時値（欠測値）は回数カウントから除外されます。
	最小値	記録周期内での最小瞬時値	
	最大値	記録周期内での最大瞬時値	
	合計	Σ瞬時値	
	瞬時値	最新の収集値	
	集計ステータス	GOOD/BAD/NOTHING/QUESTIONABLE	
差分値収集	合計	Σ（瞬時値[n]-瞬時値[n-1]） ここで n は収集回	差分合計 1 差分合計 2 どちらとも通常 Sum プロパティを割り付けます。なお，差分合計の違いについて帳票パッケージ説明書を参照してください。
			Sum プロパティ以外のプロパティを指定した場合は，SumHi デフォルト（99999999）を上限値，0 を下限値として集計します。この範囲以外は欠測値として扱われます。
	集計ステータス	GOOD/BAD/NOTHING/QUESTIONABLE	

● 締め切り処理時の集計機能について

記録周期ごとに集計したデータをもとに締め切り処理を行い，帳票全体の集計演算をすることができます。集計演算のもとになるデータは ISAM ファイル内のデータではないため，締め切り処理前に ISAM ファイル内のデータを変更しても，締め切り集計に反映されることはありません。
集計機能には以下のものがあります。

収集形式	集計演算の種類		演算規則 補足事項
実測値収集	平均値	（Σ記録値）/正常収集回数	無効な記録値（欠測値）は回数カウントから除外されます。
	最小値	記録報内での最小記録値	
	最大値	記録報内での最大記録値	
	合計	Σ 記録値	
差分値収集	合計Σ（記録値 n+ 1-記録値 n）ここで n は収集回	合計Σ （記録値 n + 1 -記録値 n）ここで n は収集回	欠測値を含む場合は,その回をスキップし,次回の記録値と前回の記録値の差分を計算します。記録周期時のような補正は行いません。通常は使用しません。

3.9.5 知っておくと便利なこと

帳票パッケージを使う上で知っておくと便利なことがいくつかあります。以下にその内容を示しますので，帳票を設計する上で参考にしてください。

● 印刷時間を遅らせることができます。

締め切り処理を行った帳票を任意のタイミングで印刷することができます。例えば，早朝 4 時に締め切り処理を行った日報を 5 時間遅れの朝 9 時に出力することもできます。但し，ロット報には印刷の遅延時間を設定することはできません。

● 帳票印字用マクロをユーザが自由に修正することができます。

例えば，校正中のデータは帳票に印刷しないなど，ASTMAC 帳票パッケージの標準機能では実現できないものを，ユーザが印刷用マクロを変更し，ISAM ファイルから読み込んだデータを編集することで帳票を出力することができるようになります。
帳票印刷用マクロはデフォルトとして[ASTMAC インストール先\]Data\Report フォルダの各タイプ別フォルダ内に[帳票ブロック名].xls で存在しています。

● 締め切り処理前の帳票データを印刷することができます。

ASTMAC の帳票パッケージは印刷用マクロと同じ内容で，画面表示用の Excel ファイルが用意されています。このファイルを起動することにより，締め切り前のデータを画面で確認することができ，また，Excel メニューの印刷を行うことにより，締め切り前のデータを帳票として出力すことができます。

● 過去の帳票を出力することができます。

● クリップボードインタフェースを用い，過去の帳票をアプリケーションフォーム VBA やデータサーバの VBA から出力することができます。

クリップボードインタフェースについては「ASTMAC 帳票パッケージ説明書（IM 34P02H01-01）」を参照してください。

3.10　トレンド

トレンド表示は，PLC で収集したデータの変化を折れ線グラフとして表示するものです。この機能を利用することにより，時間経過とともに変化するデータの傾向を視覚的に捉えることができます。
データサーバ内に存在するコントロールオブジェクトの中で，Cv 数値表現のプロパティなら，どれでもトレンド表示することが可能です。ただし，あらかじめ，トレンド表示をするための設定を行う必要があります。
画面で見るトレンドグラフとして，一度にグラフ表示できるのは，最大 8 個です。
実際の開発は，標準機能であればグラフィックビルダで設定を行うだけで，実現できます。
アプリケーション実行中に軸スケールの倍率を変更するなどの処理は，メソッドとして用意されていますので，メソッドを実行するようにします。それ以外の特殊な処理を行う場合には，イベントを使用して VBA でアプリケーションを作成する必要があります。

次にトレンド表示のサンプルを示します。
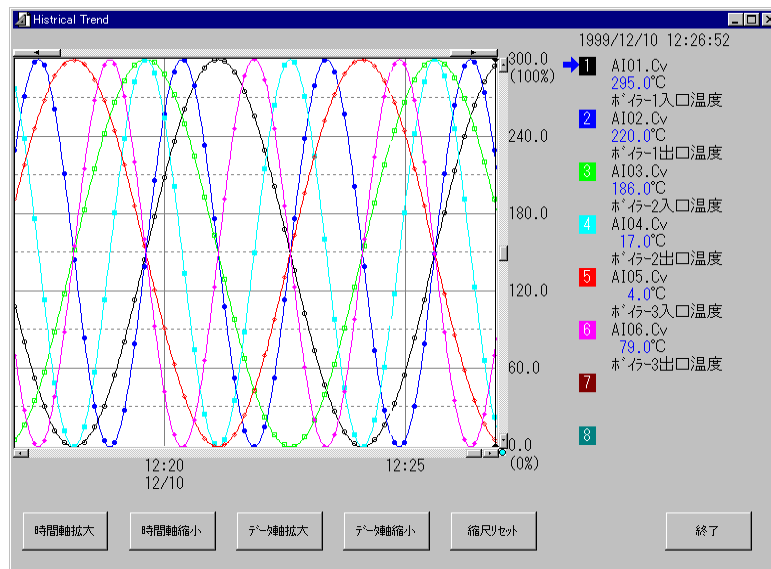
図　3.10.1 トレンド表示

上の例では，ボタンの機能は次のようになります。
時間軸拡大　：クリックするごとに時間軸が 2 倍に拡大されます。
時間軸縮小　：クリックするごとに時間軸が 1/2 倍に縮小されます。
データ軸拡大：データ軸が 2 倍に拡大されます。
データ軸縮小：データ軸が 1/2 に縮小されます。
縮尺リセット：時間軸，データ軸とも　初期状態に戻ります。

では，実際にトレンド表示パッケージを使用する場合の設計を次の項目に分けて簡単に説明します。

- トレンド種類の選択
- 表示対象となるデータの設定
- 表示画面の設計
- 設計上の注意事項

3.10.1 トレンド種類の選択

ASTMAC のトレンドパッケージには，機能を大別すると 3 種類のトレンドグラフがありますが，代表的な機能構成としては，次のようになっています。
データサーバ上のコントロールオブジェクトの値をトレンドグラフ用として定周期で収集（ヒストリデータ収集）し，画面上にグラフとして表示します。また，指定により長期保存ファイルに記録します。
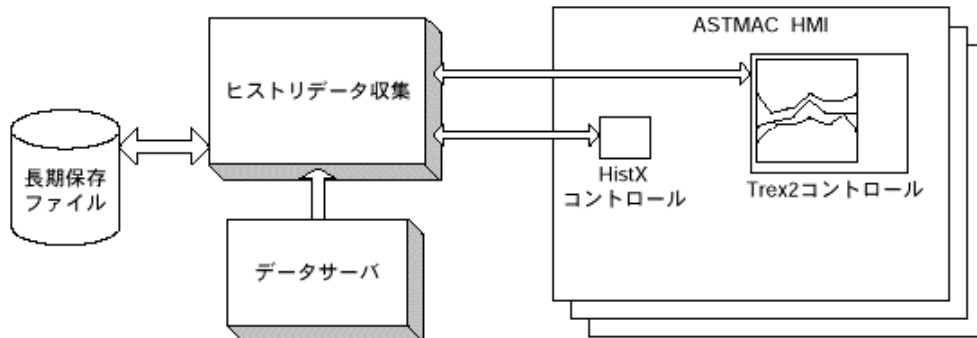
図 3.10.2 トレンドの機能構成

目的により，次の 3 種類の中からグラフを選択します。
- ヒストリカルトレンド
- リアルタイムトレンド
- バッチトレンド

ヒストリカルトレンド
　ヒストリ・データ収集を行ったデータが対象となります。
　データをファイルで保存しているので，過去のデータを任意に表示できます。

リアルタイムトレンド
　データサーバで保持しているデータが対象となります。
　データはメモリ上に保持されるため，AP 起動中のデータのみが対象になります。

バッチトレンド
　データサーバで保持しているデータが対象となります。
　データはメモリ上に保持されるため，AP 起動中のデータのみが対象になります。
　データ収集の開始／停止が制御できます。

ヒストリ・データ収集は，ヒストリグループごとに行われます。データ数の制限を以下に示します。

ヒストリグループ数	最大　32
1 つのヒストリカルグループ内のグループ数	最大　4
1 グループのオブジェクト数	最大　8

一般的には，画面上で一緒に見るオブジェクトを 1 つのグループにするように設計します。

3.10.2 表示対象となるデータの設定

ヒストリデータ収集は，最大 32 個のブロックを定義することができます。収集周期などの設定はすべてブロック単位で行います。表示対象となるデータについてブロックごとに次の項目を選択します。

● 収集周期

1 秒，2 秒，10 秒，30 秒，1 分，2 分，5 分，10 分のいずれかを選択します。
ただし，10 秒以下に設定できるのは，合計 16 ブロックまでです。

● 収集時間

5 分，10 分，30 分，1 時間，4 時間，8 時間，24 時間のいずれかを選択します。
ただし，収集周期が 10 秒以下の場合は，収集時間は 1 時間以内の中から選択します。

● 収集方式

収集方式には次の 3 つのタイプから，目的にあったものを選択します。

連続形式・ロータリー型

常時ヒストリデータを収集する方法です。システムが操業モードに移行後，自動的にデータ収集を開始します。開発モードに移行するとデータ収集も停止します。

バッチ形式・ストップ型

外部から指令を受けて，データ収集の開始や停止を行う形式です。終了指示がない場合には，記録時間いっぱいにデータを収集した時点でデータ収集が自動的に終了します。
収集開始指令と収集停止指令は以下の方法で行います。

- トレンドグラフ表示機能（TreX2 コントロール）の Collection　プロパティ
- トリガーオブジェクトの値（補足参照）
- ヒストリ収集 ActiveX コントロール（HistX コントロール）の Start メソッドと Stop メソッド

バッチ形式・ロータリー型

外部から指令を受けて，データ収集の開始や停止を行う形式です。終了指示がない場合には，記録時間いっぱいにデータを収集した時点で古いデータから順次消去し新しいデータを収集します。
収集開始指令と収集停止指令の方法は，バッチ形式・ストップ型とと同じです。

補足

トリガーオブジェクトがヒストリブロックに割り付けられていると，トリガーオブジェクトの Cv プロパティの値が 0 以外の場合に，収集が行われます。
トリガーオブジェクトには，データサーバ内の任意のコントロールオブジェクトを割り当てることができます。ただし，トリガーオブジェクトが割り当てられていると ActiveX コントロールからの Start メソッドと Stop メソッドは無効になります。
コントロールオブジェクトの Cv プロパティのみを，トリガーオブジェクトに割り付けることができます。

3.10.3 表示画面の設計

トレンド画面はグラフの部分と各データの値，名称が表示されます。データは最大8個表示できますが，縦軸は1本のみです。複数の縦軸を付けたい場合は，別に描く必要があります。
一般的には，次のような画面にします。次のトレンド表示サンプル画面は友の会ＨＰより入手できます。
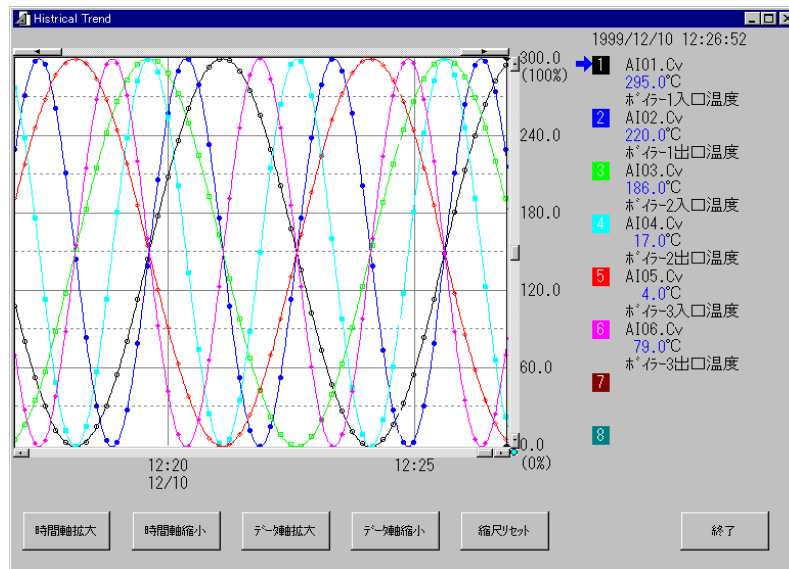
図 3.10.3 トレンド画面

この画面では，時間軸の拡大，縮小，データ軸の拡大，縮小を簡単に行うことができます。

通常は，このような画面を作成しますが，次のようなことも行うことが可能です。
・ ヒストリカルトレンドで，画面更新を行いたい場合には，UpdateRate プロパティで更新時間を設定します。
・ 既に CSV 形式で保存されたデータをトレンドグラフとして表示したい場合は，RefarenceFile プロパティで，ファイル名を指定します。

また，他にもプロパティの変更，メソッドを使用することにより種々のトレンドグラフを作成することができます。次にプロパティ，メソッド，イベントの一覧を記述しますので，参考にしてください。

プロパティ（1/2）

	名称	内容
コントロール全体	AutoAssignTag	ペンへのコントロールオブジェクトの割付を自動で行うかどうかを設定／取得します。
	AutoTagInfo	コントロールオブジェクトの情報を自動的に取得するかどうかを設定／取得します。
	BackColor	コントロールの背景色を設定／取得します。
	BlockCursorColor	ブロックカーソルの色を設定／取得します。
時間軸	ChartLength	グラフとして表示される期間を設定／取得します。
	ChartLengthUnit	表示期間の単位を設定／取得します。
	ChartStart	StartTime からグラフの左端までの時刻を設定／取得します。
	ChartStartUnit	表示基点の単位を設定／取得します。
	TimeColor	日付・時刻の色を設定／取得します。
	TimeFont	日付・時刻を表示する Font オブジェクトを設定／取得します。
	TimeDivision	時間軸罫線の間隔を設定／取得します。
	TimeDivisionUnit	時間軸罫線の間隔の単位を設定／取得します。
データ軸	DataColor	データ軸のスケールの色を設定／取得します。
	DataScaleFont	データ軸のスケールを表示する Font オブジェクトを取得します。
	MaxRange	グラフとして表示されるデータ軸の上端を設定／取得します
	MinRange	グラフとして表示されるデータ軸の下端を設定／取得します。
グラフ部分	GraphBackColor	グラフ部分の背景色を設定／取得します。
	GraphType	グラフタイプを設定／取得します。
	GridColor	データ軸罫線，時間軸罫線の色を設定／取得します。
	Heigtht	コントロールの高さを設定します。
インデックス	IndexmarkColor	インデックスマークの色を設定／取得します。
	IndexmarkTimeColor	インデックスマーク時刻の色を設定／取得します。
	IndexmarkTimeFont	インデックスマーク時刻を表示する Font オブジェクトを設定／取得します。
	IndexmarkTime	インデックスマーク位置の時刻を取得します。
ペン	Pen	アクティブなペン番号を設定／取得します。
	PenColor(n)	ペンの色を設定／取得します。
	Hide(n)	ペンを表示するかどうかを設定／取得します。
	SelectPen	選択されているペンのペン番号を設定／取得します。
タグ	TagColor	コントロールオブジェクト名，プロパティ名，コメント，瞬時値，工業単位の色を設定／取得します。
	TagComment(n)	ペンのコメントを設定／取得します。
	TagData(n)	インデックスマーク位置の時刻における，ペンの瞬時値データを取得します。デザイン時には使用できません。
	TagFont	コントロールオブジェクトの情報を表示する Font オブジェクトを設定／取得します。
	TagHighValue(n)	ペンの上限値を設定／取得します。
	TagLowValue(n)	ペンの下限値を設定／取得します。
	TagName(n)	ペンに割り付けるコントロールオブジェクトを設定／取得します。
	TagUnit(n)	ペンの工業単位を設定／取得します。
	TagWidth	コントロールオブジェクトの情報の表示幅を設定／取得します。
参照ファイル	ReferenceFile	参照パターンのファイル名を設定／取得します。
	ReferencePen(n)	参照ファイルの内容を表示するかどうか設定／取得します。
データ収集	Collection	収集の状態を設定／取得します。
	Interval	収集周期を設定／取得します。
	IntervalUnit	収集周期の単位を設定／取得します。
	Span	収集期間を設定／取得します。
	SpanUnit	収集期間の単位を設定／取得します。
	StartTime	収集開始時刻（最後の収集時刻）を設定／取得します。

プロパティ（2/2）

	名称	内容
その他	CollectionGroup	ヒストリグループを設定／取得します。
	DisplayMode	表示モードを設定／取得します。
	Enabled	ユーザ作成イベントに反応するかどうかを設定します。
	Left	フォームの左端を基点として，コントロールの左端の位置を設定します。
	Name	コード内でコントロール用に使用する名前を設定します。
	Object TreX	コントロールのオブジェクトを取得します。
	Reservation	ユーザプログラムが実行時にプロパティ変更のタイミングを設定／取得します。
	Top	コントロール内部上端からフォーム上端までの距離を設定します。
	TriggerTag	トリガータグを設定／取得します。
	UpdateRate	画面の更新時間を設定／取得します。
	UpdateRateUnit	画面の更新時間の単位を設定／取得します。
	Visible	実行時にコントロールを表示するかどうかを設定します。

イベント

	名称	内容
収集データによる	Change	収集データの内容が変更されたとき
	ChangeChart	グラフの表示範囲が変更されたとき
	ConnectionError	ヒストリデータ収集，データサーバとの間でエラーが発生したとき
	LostData	リアルタイムトレンドで全ペンのデータが欠測したとき
ユーザ操作による	EndMove	マウスポインタにより TreX2 コントロールのインデックスマークを移動が終了したとき
	GotFocus	ユーザが TreX2 コントロールのウィンドウをクリックしたり，Tab キーを押してフォーカスを移動したりして，TreX2 コントロールがフォーカスを得たとき
	LostFocus	ユーザの操作により TreX2 コントロールがフォーカスを失ったとき
	StartMove	マウスポインタにより TreX2 コントロールのインデックスマークの移動を開始したとき
その他	EndBatch	バッチトレンドの収集が終了したとき
	PenSelect	ブロックカーソルが移動したとき
	StartBatch	バッチトレンドの収集が開始されたとき

メソッド

	名称	内容
グラフ表示	ChartDown	表示範囲を下へ移動します。
	ChartLeft	表示範囲を左へ移動します。
	ChartRight	表示範囲を右へ移動します。
	ChartUp	表示範囲を上へ移動します。
データ軸	ContractDataScale	データ軸スケールの倍率を小さくします。
	ExpandDataScale	データ軸スケールの倍率を大きくします。
時間軸	ContractTimeScale	時間軸倍率を現状の倍率×1/2 にします。
	ExpandTimeScale	時間軸倍率を現状の倍率×2 にします。
インデックス	IndexmarkLeft	インデックスマークを左へ移動します。
	IndexmarkRight	インデックスマークを右へ移動します。
その他	Open Save()	メソッドで保存したトレンドグラフをロードし画面に表示します。
	Pause	リアルタイムトレンドで，画面更新の一時中断／中断の解除をします。
	ResetGraph	時間軸，データ軸，インデックスマークを初期値にリセットします。
	Save	表示中のグラフをファイルに保存します。

3.10.4 設計上の注意事項

トレンド表示を行う時のメモリ計算は次のようになります。

リアルタイムトレンドの場合
画面：表示領域サイズ Max:1024*768*24Bit（色）＝約 3MByte　－①
リアルタイムトレンド使用時保持データ
ペン数（最大 8）＊データ数（最大 10,000 点）＊データ（8Byte）
＝640Kbyte（最大）－②
（ペン数＋1）＊表示幅ドット数（最大 1024 ドット）＊4Byte
＝320Kbyte（最大）－③
上記　①＋②＋③　で求められます。

ヒストリカルトレンドの場合
画面：表示領域サイズ Max:1024*768*24Bit（色）＝約 3MByte　－①
ヒストリカルトレンド使用時保持データ
ペン数（最大 8）＊データ数（最大 32768 点）＊データ（8Byte）
＝約 2Mbyte（最大）－②
（ペン数＋1）＊表示幅ドット数（最大 1024 ドット）＊4Byte
＝320Kbyte（最大）－③
上記　①＋②＋③　で求められます。

4. テスト

実際にプログラムが完成すると，次はテストを行う必要があります。
テストとしては，単純にプログラムが正常に動作するかどうか，要求機能を実現しているか，実際の運用に耐えられるかなどのテストを行う必要があります。このようなテストを効率よく行うために，ASTMAC では「テストパッケージ」を用意しております。この章では，「テストパッケージ」を導入することを前提に，その効果的な利用方法について記述します。

以下にテストの種類と作成する書類を示します。
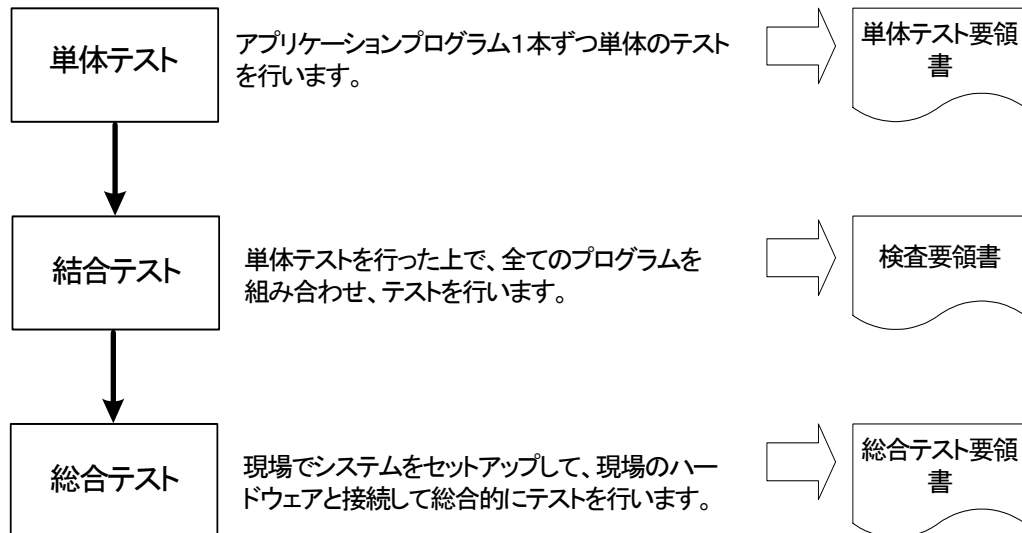
図 4.1 テストの種類と作成する書類

4.1 テストツール

ASTMAC では前記のテストを簡単に行えるようにテストツールを取り揃えております。また，Windows の機能を合わせて利用することにより効果的なテストが可能となります。

以下にテストツールを示します。

No	ツール名称	機能概要	種類	参照 IM
1	イベントビューア	ASTMAC システムのシステムメッセージは，Windows のイベントログへロギングされます。このイベントログの内容は，Windows のイベントビューアで確認することができます。	Windows	ユーザーズマニュアル
2	メッセージビューア	ASTMAC システムプログラムやアプリケーションプログラムで発生するエラー，アラームは，メッセージ履歴ファイルへロギングされ，メッセージビューアで確認できます。	ASTMAC 標準	エンジニアリングマニュアル
3	デバイスモニタ	ASTMAC のデバッグを支援するためのもので，マスタステーションに接続された PLC などのコントローラから収集したデバイス状態を画面上に表示します。	テストパッケージ	テスト機能パッケージ説明書
4	オブジェクトトレース	動作結果を記録するトレース機能です。トレース結果は，Windows のワードパットや市販のエディタ等で確認できます。	テストパッケージ	テスト機能パッケージ説明書
5	オブジェクトビューア	データサーバ上に定義されたコントロールオブジェクトの現在値を参照することができます。	ASTMAC 標準	データサーバ解説書
6	パフォーマンスモニタ	Winndows の標準機能です。システム負荷の確認ができます。	Windows	－
7	タスクマネージャ	Winndows の標準機能です。タスク状態の確認ができます。	Windows	－
8	シミュレート	マスタステーションにコントローラを接続しない状態で，マスタステーション上でコントローラのデータ変化を模擬入出力（シミュレート）します。	テストパッケージ	テスト機能パッケージ説明書

では，実際にどのようなテストを行うときに利用するか，機能をベースに説明します。
機能ごとのテストとして以下の項目があります。

- データサーバの確認
- 操作／監視画面の確認
- アラーム／メッセージの確認
- 機能確認

● データサーバの確認

データサーバの確認としては，PLC とのデータ交換の確認とデータ処理の確認を行います。

① I/O オブジェクトの確認
データサーバとコントローラとの接点である I/O オブジェクトの確認を行います。
「デバイスモニタ」で I/O オブジェクトの各デバイスを表示させ，コントローラ側のデバイスと内容が合っているかを確認します。
また，PLC とのタイミングによるプログラムの動きを確認する場合に，デバイスデータを直接変更し，マスタステーションのプログラムと PLC のプログラムの動作確認を行います。

② データ処理の確認
データ処理には，開始タイミングと処理結果の確認を行う必要があります。
開始タイミングで使用しているコントロールオブジェクトは，「オブジェクトビューア」で確認できます。処理結果の確認は，結果を表示する画面やファイルの値を確認します。
アプリケーションが正常に動作しない場合に，「オブジェクトトレース」を使用して原因を追求することもできます。

● 操作／監視画面の確認

ここでは画面展開，各画面のモニタ表示内容，そして操作時の他のタスクへの反映を確認します。

① コントローラからの取得データを確認
テスト機能パッケージの「デバイスモニタ」を使用して，表示されているデータの値を確認するため，レジスタの数値変更，リレーの ON/OFF 切り替え等を行い，画面に表示されている値とデバイスモニタの値が一致することを確認します。

② 画面からの操作
PC の画面でのデータ設定，リレー等の ON/OFF 切り替えを行う場合は，変更した内容を①と同様に「デバイスモニタ」で確認します。

● アラーム／メッセージの確認

① プロセスアラーム／メッセージの確認
テスト機能パッケージの「デバイスモニタ」を使用して，デバイスの値を変化させ，意図的にアラーム，メッセージを発生させます。アラームやメッセージが発生することをメッセージビューアの画面で確認できます。

② ユーザアラーム／メッセージの確認
アプリケーションで定義したアラームやメッセージが発生するような状況を意図的に作り出して確認します。そのために，「デバイスモニタ」を使用して行うのも効果的です。

● 機能確認

既設のコントローラと接続する場合など，実際にコントローラなしで ASTMAC システムの確認をしなければいけないような場合があります。そのような場合に，簡単に模擬入力できる機能を持った「シミュレート」が用意されています。
ASTMAC の「シミュレート」は，単体テストなどでアプリケーションプログラムのデバッグをする場合に有効です。シミュレート機能は，マスタステーションにコントローラを接続しない状態で，マスタステーション上でコントローラのデータ変化を模擬入出力（シミュレート）することができます。

「シミュレート」は，デバイスオブジェクト単位で行い，次の種類があります。
デジタル入力，デジタル出力，デジタル入出力，アナログ入力，アナログ出力，アナログ入出力，アナログ入力，拡張アナログ出力，拡張アナログ入出力

シミュレートを行うためにシミュレートデータを設定しますが，シミュレートデータには，定型パターンとユーザ定義パターンの 2 種類があります。また運用時に直接シミュレートデータの変更を行うことができます。

・ 定型パターン
  サイン波などの決まった波形を繰り返し発生します。

・ ユーザ定義パターン
  ユーザが定義したシミュレートのパターンを出力します。

・ 手動による変更
  シミュレートデータを任意に，直接変更することができます。

シミュレートにて機能ごとのテストを行い，正常に動作しなかった場合，デバイスモニタなどでデータを確認するとともに，トレース機能を使用してプログラムの動きまたはデータの変化を確認することができます。

トレース結果は CSV 形式ファイルとして記録されます。
Windows のワードパットや市販のエディタ等でファイルを開いて，データを確認します。

また，アプリケーションソフトが反応しないような場合は，Windows のタスクマネージャおよびパフォーマンスモニタを使用して，アプリケーションソフトの稼動状態，システムの負荷率を確認できます。

5. 付録
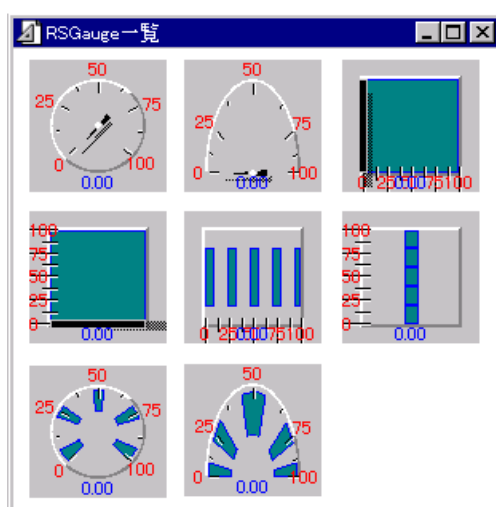
図 5.1 ActiveX コントロール（RSGauge）
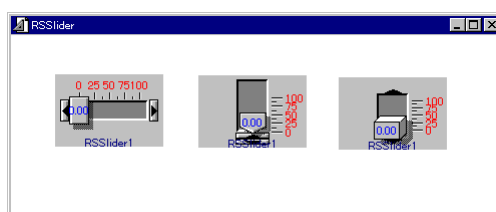
図 5.2 ActiveX コントロール（RSSlider）
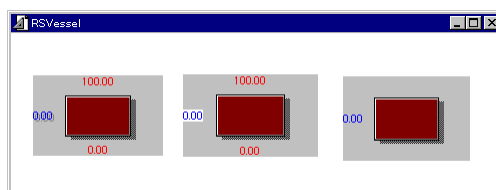
図 5.3 ActiveX コントロール（RSVessel）
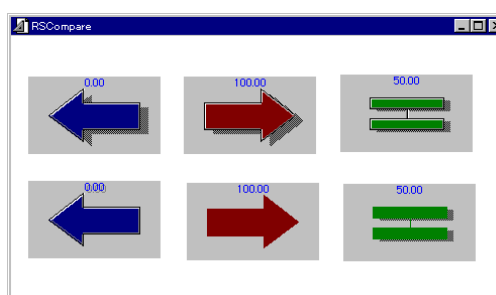
図 5.4 ActiveX コントロール（RSCompare）
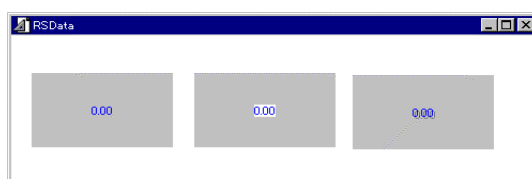
図 5.5 ActiveX コントロール（RSData）
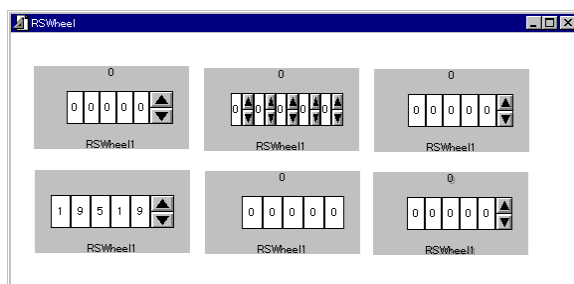
図　5.6 ActiveX コントロール（RSWheel）
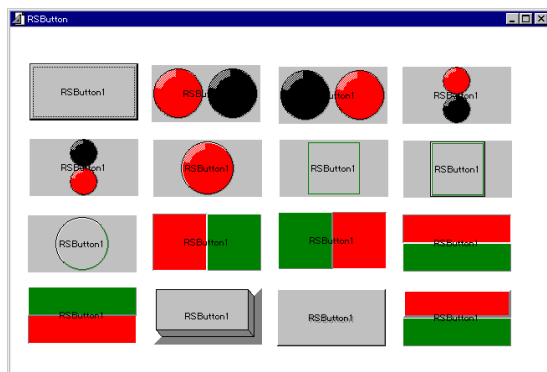
図　5.7 ActiveX コントロール（RSButton）
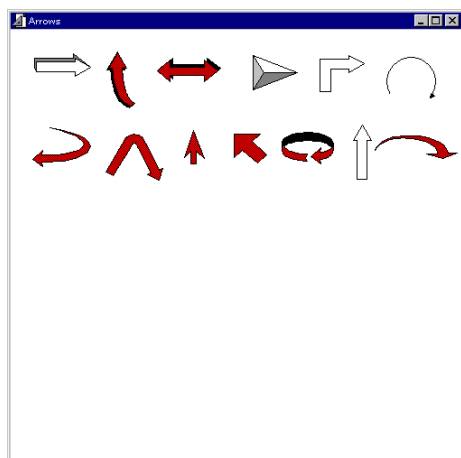
図　5.8 シンボルライブラリ（Arrows.odf）
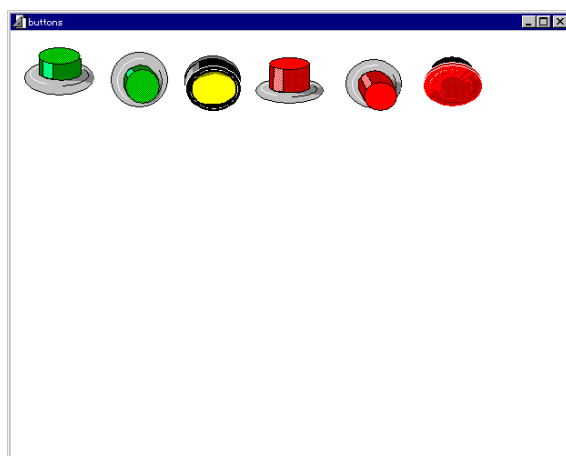
図　5.9 シンボルライブラリ（Buttons.odf）
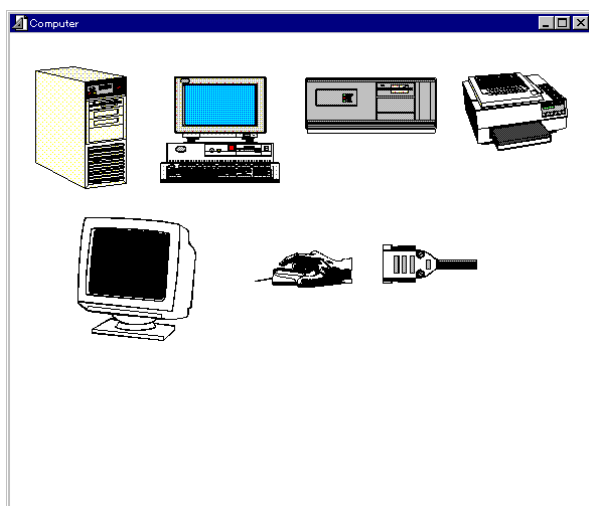
図　5.10 シンボルライブラリ（Computer.odf）
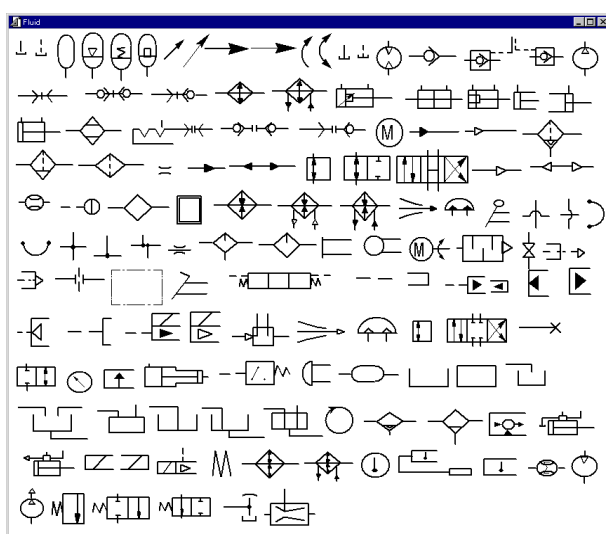
図　5.11 シンボルライブラリ（Fluid.odf）
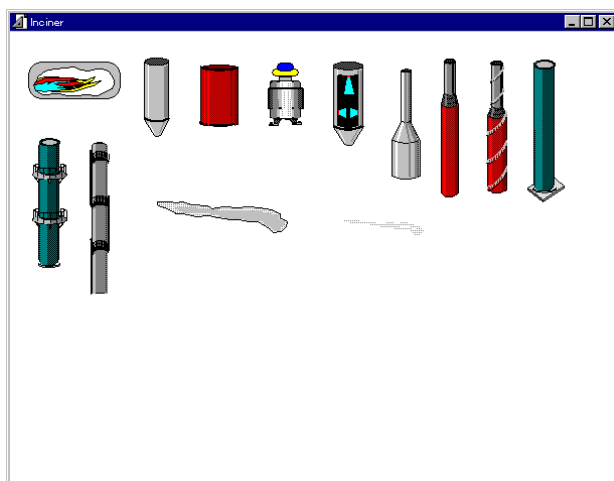
図　5.12 シンボルライブラリ（Inciner.odf）
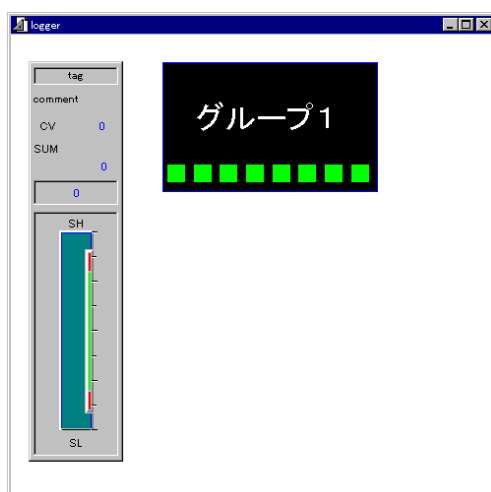
図　5.13 シンボルライブラリ（Logger.odf）
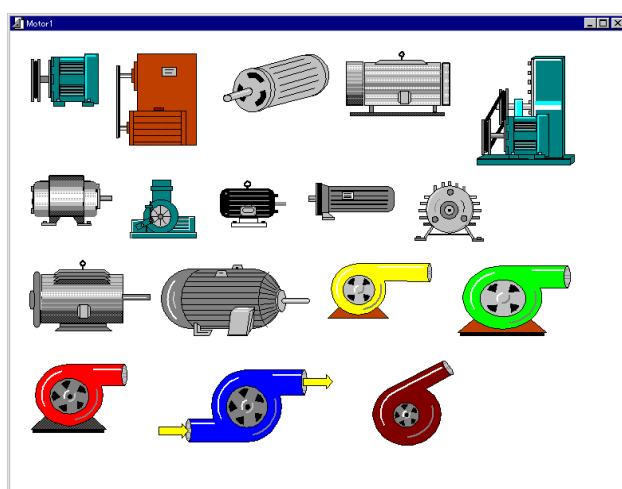
図　5.14 シンボルライブラリ（Motor1.odf）
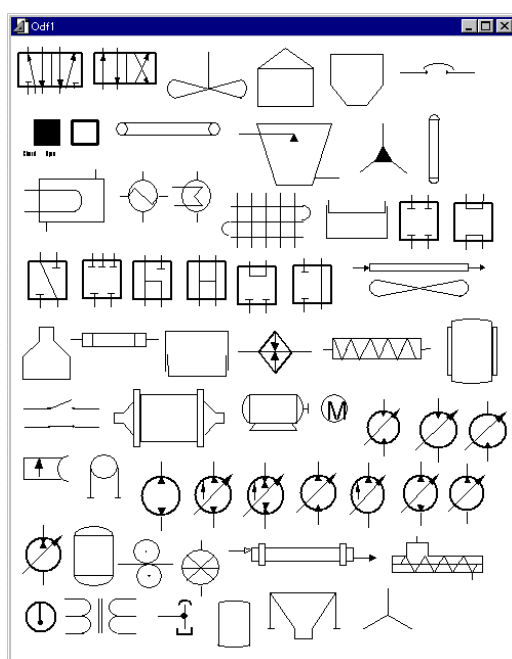
図　5.15 シンボルライブラリ（Odf1.odf）
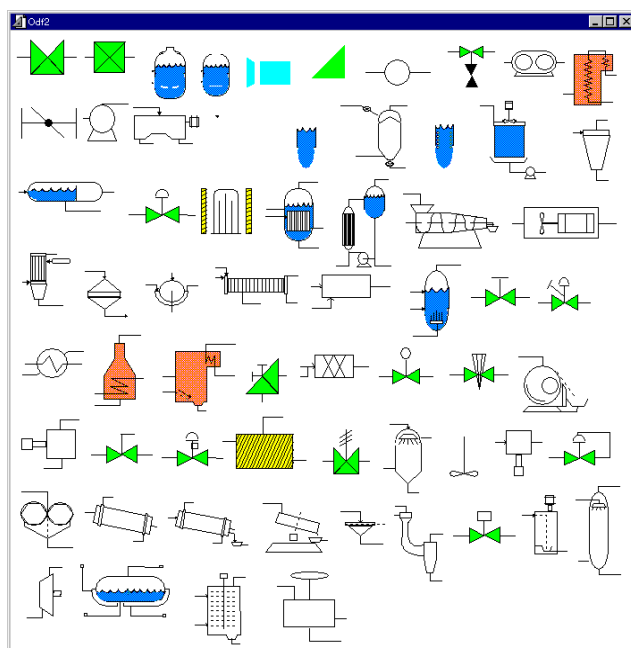
図　5.16 シンボルライブラリ（Odf2.odf）
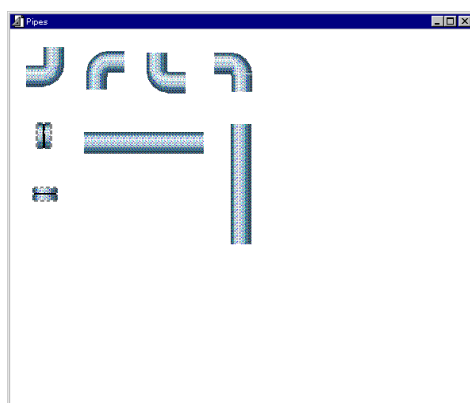
図　5.17 シンボルライブラリ（Pipes.odf）
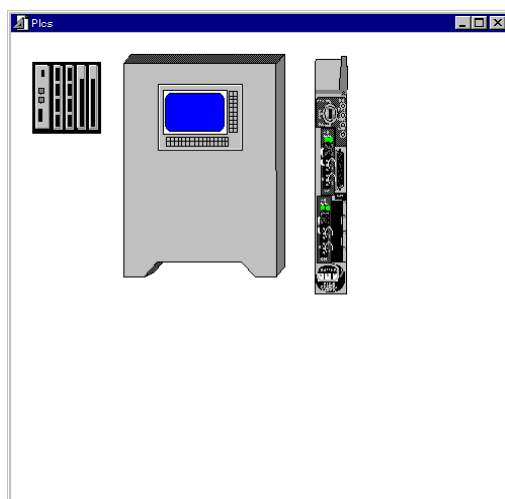
図　5.18 シンボルライブラリ（Plcs.odf）
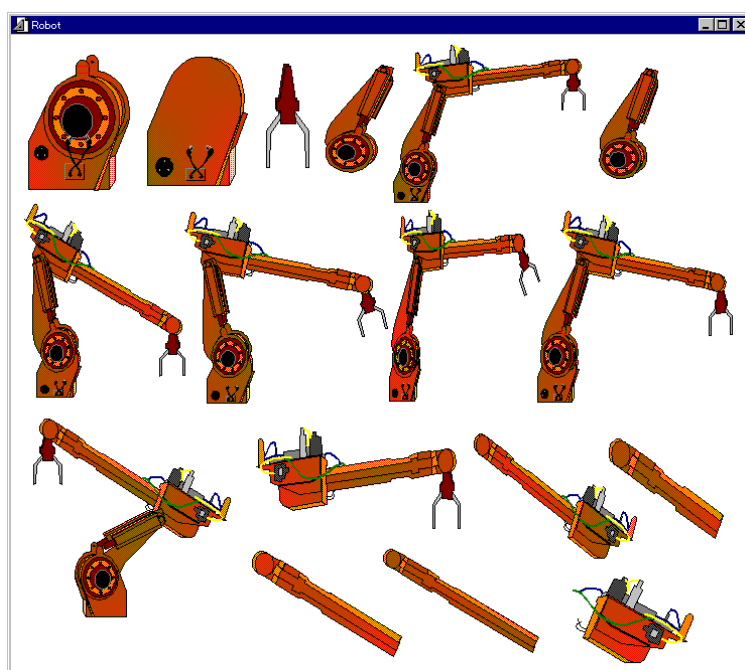
図　5.19 シンボルライブラリ（Robot.odf）

図　5.20 シンボルライブラリ（Signs1.odf）
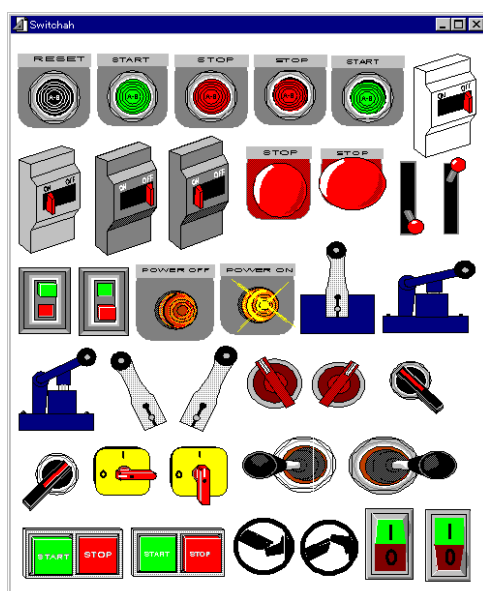
図　5.21 シンボルライブラリ（Switchah.odf）
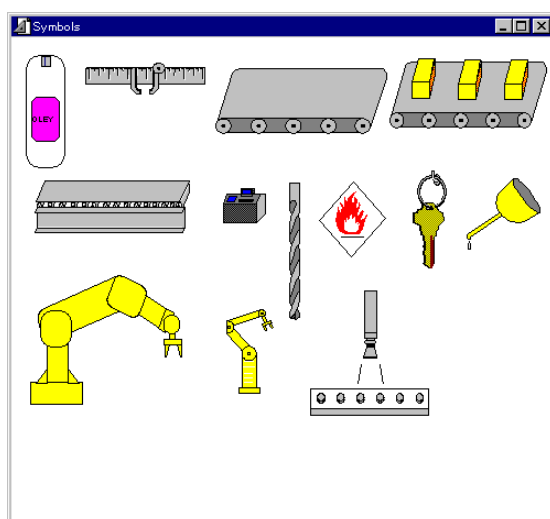
図　5.22 シンボルライブラリ（Symbols.odf）
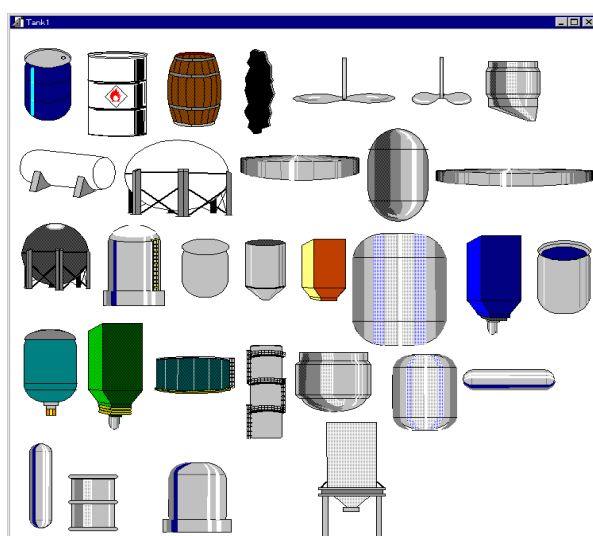
図　5.23 シンボルライブラリ（Tank1.odf）
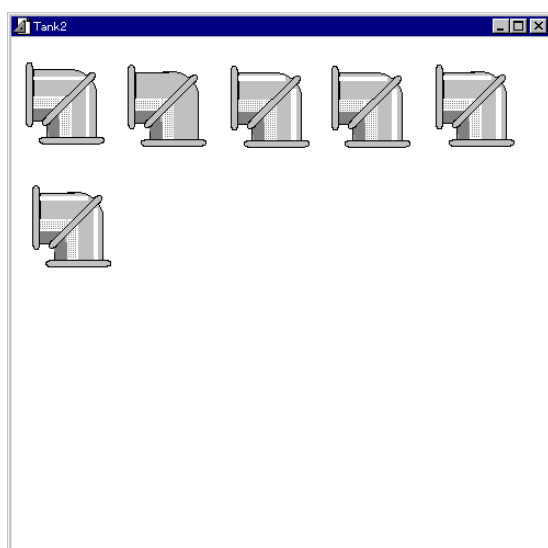
図　5.24 シンボルライブラリ（Tank2.odf）
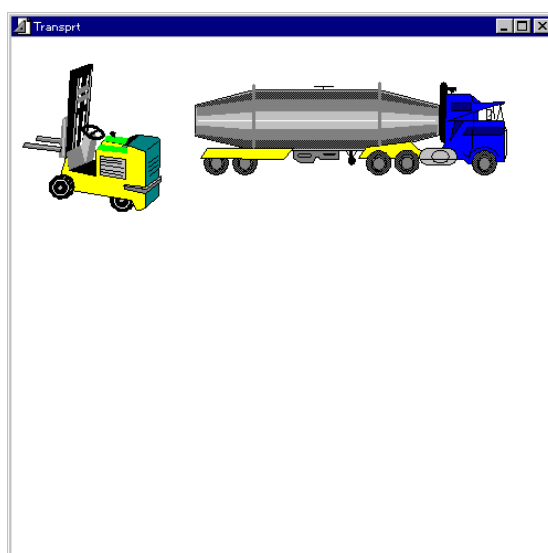
図　5.25 シンボルライブラリ（Transprt.odf）
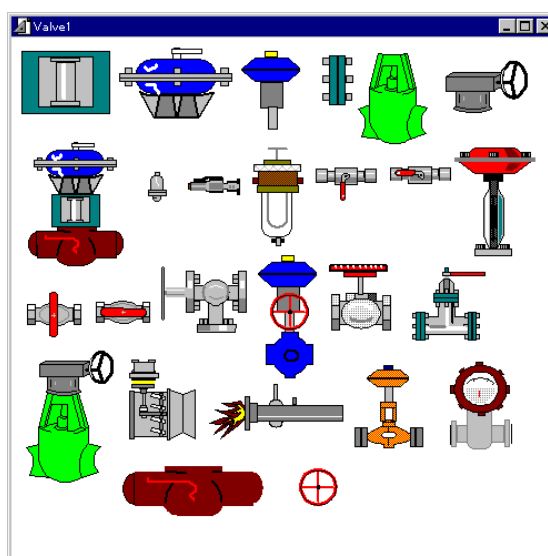
図　5.26 シンボルライブラリ（Valve1.odf）
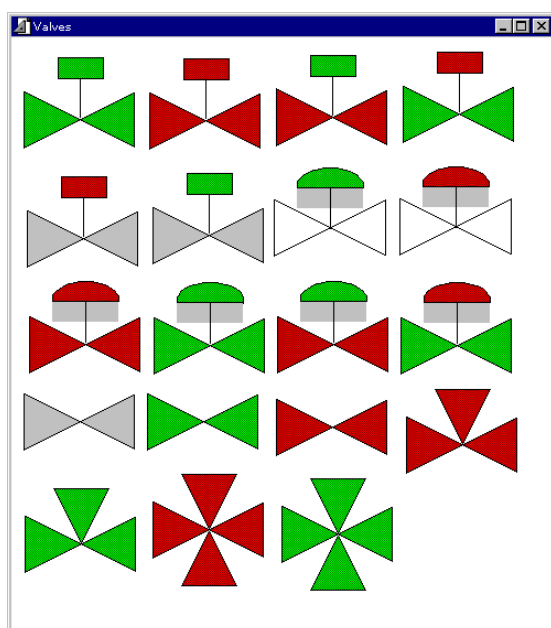
図　5.27 シンボルライブラリ（Valves.odf）
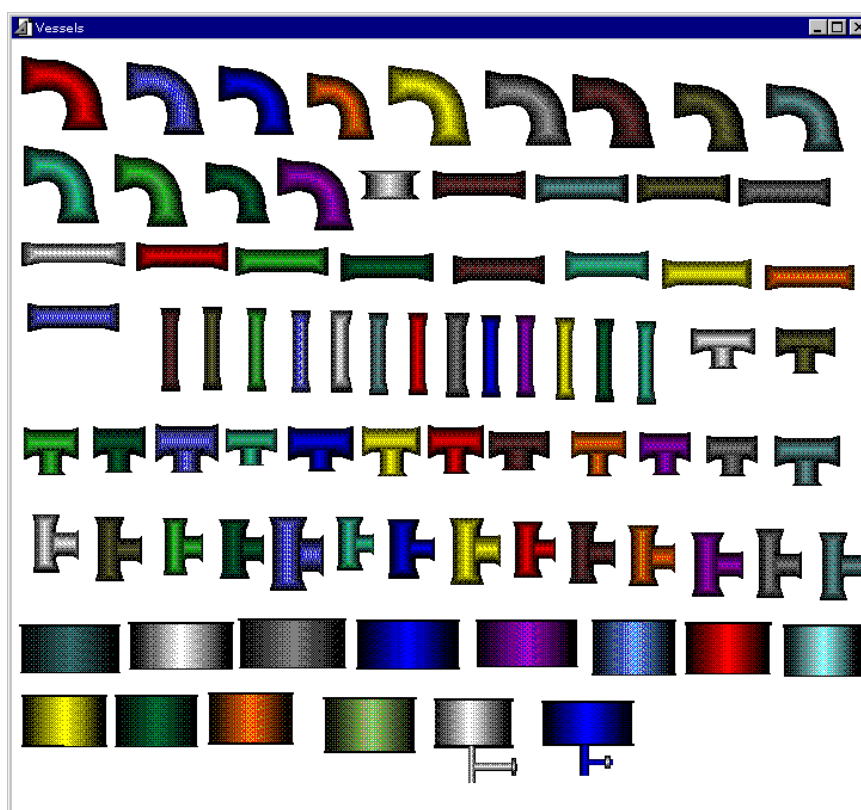
図　5.28 シンボルライブラリ（Vessels.odf）

Technical Information　改版履歴

資料名称：　ASTMAC エンジニアリングガイド
資料番号：　TI 34P02K02-01

2000 年 1 月／初版／R2.05 以降
新規発行

2011 年 4 月／2 版／R7.01 以降*
改訂
・R7.01 対応（Windows 7 対応など）
・誤記訂正

*：Technical Information 記載内容と対応しているソフトウェアのリリース番号。対応する範囲は次の改訂版が発行されるまで。

著作者　横河電機株式会社
IA マーケティングセンター
PAPMK 部
発行者　横河電機株式会社
〒180-8750　東京都武蔵野市中町 2-9-32